Université de Montréal

Traffic Prediction and Bilevel Network Design

par

Léonard Ryo Morin

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée en vue de l'obtention du grade de
Philosophiæ Doctor (Ph.D.)
en Informatique

10 janvier 2020

Université de Montréal

Faculté des études supérieures et postdoctorales

Cette thèse intitulée

Traffic Prediction and Bilevel Network Design

présentée par

Léonard Ryo Morin

a été évaluée par un jury composé des personnes suivantes :

Jean-Yves Potvin
(président-rapporteur)

Emma Frejinger
(directrice de recherche)

Bernard Gendron
(codirecteur)

Fabian Bastin
(codirecteur)

Margarida Carvalho
(membre du jury)

Carolina Osorio
(examinatrice externe)

Jacques Bélair
(représentant du doyen de la FESP)

Résumé

Cette thèse porte sur la modélisation du trafic dans les réseaux routiers et comment celle-ci est intégrée dans des modèles d'optimisation. Ces deux sujets ont évolué de manière plutôt disjointe: le trafic est prédit par des modèles mathématiques de plus en plus complexes, mais ce progrès n'a pas été incorporé dans les modèles de design de réseau dans lesquels les usagers de la route jouent un rôle crucial. Le but de cet ouvrage est d'intégrer des modèles d'utilités aléatoires calibrés avec de vraies données dans certains modèles biniveaux d'optimisation et ce, par une décomposition de Benders efficace. Cette décomposition particulière s'avère être généralisable par rapport à une grande classe de problèmes communs dans la litérature et permet d'en résoudre des exemples de grande taille.

Le premier article présente une méthodologie générale pour utiliser des données GPS d'une flotte de véhicules afin d'estimer les paramètres d'un modèle de demande dit recursive logit. Les traces GPS sont d'abord associées aux liens d'un réseau à l'aide d'un algorithme tenant compte de plusieurs facteurs. Les chemins formés par ces suites de liens et leurs caractéristiques sont utilisés afin d'estimer les paramètres d'un modèle de choix. Ces paramètres représentent la perception qu'ont les usagers de chacune de ces caractéristiques par rapport au choix de leur chemin. Les données utilisées dans cet article proviennent des véhicules appartenant à plusieurs compagnies de transport opérant principalement dans la région de Montréal.

Le deuxième article aborde l'intégration d'un modèle de choix de chemin avec utilités aléatoires dans une nouvelle formulation biniveau pour le problème de capture de flot de trafic. Le modèle proposé permet de représenter différents comportements des usagers par rapport à leur choix de chemin en définissant les utilités d'arcs appropriées. Ces utilités sont stochastiques ce qui contribue d'autant plus à capturer un comportement réaliste des usagers. Le modèle biniveau est rendu linéaire à travers l'ajout d'un terme lagrangien basé sur la dualité forte et ceci mène à une décomposition de Benders particulièrement efficace. Les expériences numériques sont principalement menés sur un réseau représentant la ville de Winnipeg ce qui démontre la possibilité de résoudre des problèmes de taille relativement grande.

Le troisième article démontre que l'approche du second article peut s'appliquer à une forme particulière de modèles biniveaux qui comprennent plusieurs problèmes différents. La décomposition est d'abord présentée dans un cadre général, puis dans un contexte où le second niveau du modèle biniveau est un problème de plus courts chemins. Afin d'établir que ce contexte inclut plusieurs applications, deux applications distinctes sont adaptées à la forme requise: le transport de matières dangeureuses et la capture de flot de trafic déterministe. Une troisième application, la conception et l'établissement de prix de réseau simultanés, est aussi présentée de manière similaire à l'Annexe B de cette thèse.

Mot clés: données GPS, choix de chemin, modèles récursifs de choix, terminaux intermodaux, capture de flot, décomposition de Benders, optimisation biniveau, maximisation d'utilité aléatoire.

Abstract

The subject of this thesis is the modeling of traffic in road networks and its integration in optimization models. In the literature, these two topics have to a large extent evolved independently: traffic is predicted more accurately by increasingly complex mathematical models, but this progress has not been incorporated in network design models where road users play a crucial role. The goal of this work is to integrate random utility models calibrated with real data into bilevel optimization models through an efficient Benders decomposition. This particular decomposition generalizes to a wide class of problems commonly found in the literature and can be used to solved large-scale instances.

The first article presents a general methodology to use GPS data gathered from a fleet of vehicles to estimate the parameters of a recursive logit demand model. The GPS traces are first matched to the arcs of a network through an algorithm taking into account various factors. The paths resulting from these sequences of arcs, along with their characteristics, are used to estimate parameters of a choice model. The parameters represent users' perception of each of these characteristics in regards to their path choice behaviour. The data used in this article comes from trucks used by a number of transportation companies operating mainly in the Montreal region.

The second article addresses the integration of a random utility maximization model in a new bilevel formulation for the general flow capture problem. The proposed model allows for a representation of different user behaviors in regards to their path choice by defining appropriate arc utilities. These arc utilities are stochastic which further contributes in capturing real user behavior. This bilevel model is linearized through the inclusion of a Lagrangian term based on strong duality which paves the way for a particularly efficient Benders decomposition. The numerical experiments are mostly conducted on a network representing the city of Winnipeg which demonstrates the ability to solve problems of a relatively large size.

The third article illustrates how the approach used in the second article can be generalized to a particular form of bilevel models which encompasses many different problems. The decomposition is first presented in a general setting and subsequently in a context where the lower level of the bilevel model is a shortest path problem. In order to demonstrate that

this form is general, two distinct applications are adapted to fit the required form: hazmat transportation network design and general flow capture. A third application, joint network design and pricing, is also similarly explored in Appendix B of this thesis.

Keywords: GPS data, route choice, recursive choice models, intermodal terminals, flow capture, Benders decomposition, bilevel optimization, random utility maximization.

Contents

List of Tables

List of Figures

List of Acronyms and Abbreviations

GPS	Global Positioning System
GMA	Greater Montreal Area
OD	Origin-Destination
FCP	Flow Capture Problem
RUM	Random Utility Maximization
LP	Linear Program
MIP	Mixed Integer Program
MILP	Mixed Integer Linear Program
$\mathcal{CS}$	Complementary Slackness
$\mathcal{SD}$	Strong Duality
$\mathcal{L}$	Lagrangian
BD	Benders Decomposition
ME	Mildly Evasive
VE	Very Evasive
MC	Mildly Cooperative
VC	Very Cooperative

HTNDP	Hazmat Transportation Network Design Problem
JNDP	Joint Network Design and Pricing

Acknowledgements

First and foremost, I would like to thank my research supervisor Emma Frejinger for guiding me throughout my doctoral studies. Her unwavering diligence greatly contributed to the quality of this work. Her words of encouragement allowed me to persevere through the numerous setbacks that befell our research. Above all however, her kindness and concern for my interests have made the past few years a time I will always look back on fondly.

I am also very grateful to my co-supervisor Bernard Gendron whose expertise provided the impetus behind many developments found in this thesis. His relentless curiosity and rigorous scrutiny of our experimental results gave us valuable insights which further strengthened our research.

I would like to express my thanks to Fabian Bastin as well. In addition to providing very useful feedback in regards to the first article of this thesis, he was the person who originally suggested a PhD with Emma.

Lastly, I must thank my family and friends. Surrounded by them, my life outside of my studies was just as enjoyable as my pursuit of a doctoral degree.

Introduction

This thesis covers the three main following topics: demand model parameter estimation using real GPS data, integrating random utility models into a general flow capture problem and a Benders decomposition specially adapted to a broad class of bilevel models. Throughout this introduction, it is our goal to not only show how these seemingly unrelated subjects lead one into the other, but also the importance of the whole they form. The chapter is structured as follows. First, we showcase the motivation behind this work. Second, we provide a background to the main areas of our research. Third, we lay out our specific objectives and the resulting contributions. Finally, we detail the outline for the rest of this thesis.

Motivation

With the global population more than tripling since the beginning of the 20th century, a need for efficient methods of planning, decision making and simulation came into prominence. This plight was answered by the development of operations research. Owing to the advent of the information age, countless works of ever increasing complexity have been published in a myriad of fields. One of these fields is the study of demand modeling which can be applied to characterizing the behavior of people travelling by choosing one of many alternatives. People and their movements are at the crux of many services, phenomenons and issues such as public transit, air travel, the rise of alternative fuel vehicles, traffic congestion, the concomitant CO_2 emissions and other negative impacts. It is thus imperative to accurately portray the behavior of these people if we are to enact effective policies and take actions entailing environmental and economic benefits. Tracking people with GPS has emerged, in recent years, as a powerful means to that end because of both its quantity and availability. Therefore, providing a way to harness this plentiful resource with the goal of grasping the factors involved in our decision-making processes in regards to our travel choices is of the utmost importance.

However, this undertaking only serves as a basis for determining the choices we make relating to any problematic situation we wish to address. In mathematical programming, decisions are reached by the coalescence of an objective and constraints of which demand

modeling is often a pivotal element. What is typically used in that regard can be greatly improved by simply incorporating state of the art demand models which have evolved separately. Hence, it beckons to us to unite them with a problem where the behavior of travellers is a central theme. We choose the general flow capture problem as we believe it is a peculiarly versatile setting with numerous concrete applications due to its nature: an authority decides where to install certain flow capturing resources, with the aim of capturing as much traffic as possible, in a network where users, who have a reaction to these resources, are travelling. The essence of this situation revolves around the behavior of the travellers and their response to the decisions made by the authority, which, together, dictate the optimal installation configuration. Consequently, incorporating the best representation of the behavior of these travellers in our mathematical program is fundamental.

With a mathematical model able to accurately predict the choices of travellers, the last hurdle which we must clear is that of scaling. Indeed, the unprecedented growth of humanity requires us to consider ever expanding settings which, in turn, translate to extremely large mathematical models. A common approach to circumventing this impediment is the use of heuristics which simplify reality to more manageable dimensions. However, the aptly named decomposition methods deal with the issue by breaking down the problem into smaller parts which can then be addressed efficiently. It is in fact what we use for the general flow capture problem and we demonstrate that our method can be applied to a wide variety of other problems. Thus, the goal is to provide the means for the models used to solve all of these problems to be able to scale up to realistic proportions while simultaneously accurately representing travellers' behaviors which allows for better decisions.

Research Background

This thesis covers several fields of research and, as such, only the most relevant to this work are discussed here. We also aim to provide, when appropriate, a perspective on what we consider to be a lack of connection between these various topics in the state of the art. Of course, a more thorough overview can be found in the literature review section of each article presented subsequently.

We first look at discrete choice modeling in the context of the path choice of individuals travelling within a network of nodes and arcs. Largely owing their popularity to the seminal work of [54], discrete choice models are used to predict the choices of an individual when choosing from a discrete set of alternatives. This choice is assumed to be based on attributes captured by the utility of the alternatives and characteristics of the individual who is utility maximizing. In reality, for various practical reasons, some of these attributes are unknown to the modeler which leads to random utility discrete choice models. These models determine choice probabilities for each alternative. The probabilities depend on assumptions made regarding the choice set and the random portion of the utility which leads to different models.

This framework naturally lends itself to describe the route choices of individuals travelling within a transportation network. This route choice can be broken down into a series of arc choices leading an individual from his origin to his destination or it can be made from a set of predetermined paths, potentially determined by choice set generation algorithms [e.g, 5, 58].

Both approaches have their strengths and weaknesses. Path-based models have the advantage of being estimated easily by commercial software. They are not conducive to reliable scenario analysis because choice probabilities depend on generated choice sets. These choice sets also have to be recalculated when changes are made to the network. Arc-based models, on the other hand, do not require path choice sets and thus do not suffer from their associated drawbacks. However, they can be limited in their ability to represent certain attributes that are specific to a complete path such as a habitual (preferred) path. More generally, arc-based models require arc-additive attributes which can be a drawback. Nevertheless, it is a widely accepted assumption with respect to shortest path calculations and it is, in fact, often used to generate path choice sets for path-based models. A more detailed overview of the two approaches can be found in [26]. For these reasons, we focus on arc-based models throughout this thesis.

The instantaneous utility of link $a \in A(k)$ is $u(a|k;\beta) = v(a|k;\beta) + \varepsilon(a)$, where $A(k)$ is the set of outgoing links from link k, $v(a|k;\beta)$ is the deterministic utility and β is a vector of parameters. These parameters typically include the travel time, the speed limit, the road safety, the scenery, etc. The $\varepsilon(a)$ represents the random term. These can be assumed to be, for example, independently and identically distributed extreme value (with location zero and scale μ) so that the choice model at each choice stage is logit. One of the most important aspects to consider is the correlation between alternatives. In our context, this translates to different possible paths having certain arcs in common. For instance, a standard logit model does not consider this correlation when calculating path choice probabilities. There are variants of the logit model that address correlation through a correction of the deterministic utilities, such as C-logit [16] and path size logit [6]. However, they rely on a complete path enumeration or choice set sampling [24]. If we define p as a sequence of arcs a_1,a_2,...,a_T, we can define its associated utility as

$$u(p) = \sum_{i=2}^{T} u(a_i|a_{i-1};\beta). \tag{1.0}$$

The probability of choosing path p would thus be given by:

$$\text{Prob}(p) = \frac{e^{v(p)}}{\sum\limits_{p' \in P} e^{v(p')}}, \tag{1.1}$$

where P is the set of all paths.

We now turn our attention to random utility models in supply-side mathematical programs. Supply-side problems, in the context of operations research, refer to problems where a manager typically has choices to make in regards to providing services, constructing facilities, allocating resources and so on while taking into account the demand. In many cases, discrete choice models that would realistically model the demand are ignored in favor of more simplistic representations. This is mainly due to the non-linear expressions for the choice probabilities of even the simplest discrete choice models as can be seen in Equation (1.1).

Keeping a mathematical program linear is an advantage as it allows to apply the more common solution techniques and thus incorporating choice probability expressions is challenging. These non-linear terms can be approximated in a number of ways: linear functions, piecewise linear functions, or even constants as a compromise [28]. There are also recent developments in conic programming [62]. There is, however, a different and recently developed approach to integrating discrete choice models in mathematical programs which is based on the utilities rather than the probabilities of each alternative. This thesis adopts this approach.

Lastly, we provide a brief overview of bilevel programming as it is one of the focal points in this work. Bilevel problems consist of a leader's problem and a follower's problem [17]. The leader is trying to optimize its own objective while respecting certain constraints while the follower does the same with its own objective and constraints as can be observed in (1.2)-(1.5) from [17]. Generally, the decisions of the leader affect the decisions of the follower by changing its objective or constraints. The new decisions of the follower can in turn affect the decisions of the leader. This cycle makes bilevel programs difficult to solve.

$$\min_{x,y} F(x,y), \tag{1.2}$$

$$\text{s.t.}\, G(x,y) \leq 0, \tag{1.3}$$

$$\min_{x} f(x,y), \tag{1.4}$$

$$\text{s.t.}\, g(x,y) \leq 0. \tag{1.5}$$

There are many types of solution methods used to deal with bilevel programs [19]. In this thesis, we focus on those that reformulate the bilevel model into an equivalent single-level model through the use of duality theory. This is usually achieved by one of two ways. The first replaces the objective function of the follower's problem with dual feasibility constraints and complementary slackness conditions. The second substitutes the complementary slackness conditions with the strong duality constraint which states that the primal and dual objective functions of a problem are equal at optimality (if a bounded optimal solution exists).

While both of these methods yield mathematical programs that can be solved by standard optimization tools, they have characteristics that may be considered problematic in certain circumstances. Working with complementary slackness conditions means adding a relatively large number of constraints that also have to be linearized which often implies adding a set of binary variables. Although it is a single constraint, the strong duality constraint can have the downside of providing the model with a relatively weak relaxation. Also, in both cases, the structure of the follower's problem cannot be exactly extracted for a decomposition method because of the additional constraints. However, there is much to gain if we were able to do so. In the context of bilevel network design problems or any other bilevel problem where the follower's problem consists of finding the shortest path in a network, shortest path problems could be solved very efficiently by algorithms such as the Dijkstra one. This thesis attempts to find a way to obtain a single-level formulation of a bilevel model which would still allow for a decomposition method to benefit from the initial structure of the follower's problem.

However, simply solving shortest path problems assumes that the followers have perfect knowledge of network attributes and minimize the same objective function. Assuming an uncongested network, for a given origin destination pair, they would hence seek the same shortest path. We can avoid this oversimplification by using the aforementioned advanced discrete choice models. Therefore, the method we use to reformulate a bilevel program to a single-level must be able to use these demand models and preserve the shortest path problem structure so that algorithms can be employed.

Objectives and Contributions

In this section, we explicitly lay out the contributions of this thesis. They are grouped by articles.

The first article, "A GPS-based Recursive Logit Model for Truck Route Choice in an Urban Area", is centered on calibrating a discrete choice model using real GPS data. The discrete choice model used is a recursive logit model which does not require choice set generation. The contribution is empirical, illustrating how to transition from GPS traces with limited data processing to an estimated set of parameters for the recursive logit model. Furthermore, because the GPS data and the road network used are genuine, the numerical results themselves have value for practical purposes.

The second article, "Flow Capture under Heterogeneous User Behavior in Uncongested Networks", introduces a new formulation for the flow capture problem. This formulation can accommodate various discrete choice models to represent how traffic propagates in the network. The formulation itself is an important contribution but the key is how a utility simulation approach is adapted to a network context. This approach relies on considering a sufficient number of realizations of arc utilities rather than non-linear probability expressions. Another important contribution in this paper is a novel Benders decomposition which

exploits the shortest path problem formulation in the follower's problem. This is possible because of the particular way in which the bilevel model is brought to a single level without complementary slackness constraints nor the strong duality constraint. The last contribution of this article takes the form of a numerical comparison between the various models developed by the more traditional complementary slackness constraints and strong duality constraint, as well as our Benders decomposition.

The third article, "Benders Decomposition for a Class of Bilevel Programs with Applications to Network Design", generalizes the decomposition method applied to the flow capture problem in the second article. The first contribution here is the general bilevel model along with the assumptions needed in order to apply our Benders decomposition. This generic model is made more specific by detailing a shortest path problem as the follower's problem. This allows us to highlight how that structure can be exploited to achieve efficient solution methods. Subsequently, we demonstrate how various bilevel problems can be adapted to fit the form needed for our decomposition. This is done through three concrete examples for which we provide a detailed transition.

To summarize, we first present an application focused on predicting truck route choices using GPS data and the recursive logit model. We then show how it can be adapted into a random utility model used in a bilevel formulation for the general flow capture problem. The resulting formulation can be relatively large, but we demonstrate how a novel Benders decomposition method can be applied to efficiently solve it. Finally, we illustrate that our decomposition can be applied to a wider class of bilevel models dealing with network design and users seeking shortest paths within a transportation network.

First Article.

A GPS-based Recursive Logit Model for Truck Route Choice in an Urban Area

by

Léonard Ryo Morin[1], Emma Frejinger[1], Fabian Bastin[1], and Martin Trépanier[2]

([1]) Department of Computer Science and Operations Research and CIRRELT
Université de Montréal, Canada

([2]) Department of Mathematics and Industrial Engineering and CIRRELT
Polytechnique Montréal, Canada

This article was published as a chapter in Sustainable City Logistics Planning: Methods and Applications, Volume 3. The data analysis project (partly reported in this paper) conducted in collaboration with CargoM, Transport Canada, CIRRELT and several freight carriers was awarded the "Grand prix d'excellence en transport: transport de marchandises" from the Association québécoise des transports (AQTr) in 2017.

My contributions for this paper include handling the data, analyzing it, producing the results and writing the text. Only the code used to estimate the demand model was not mine. Emma Frejinger helped devise the analysis and revised the writing. Fabian Bastin and Martin Trépanier contributed to the writing and provided some useful comments and references.

RÉSUMÉ. Nous explorons l'usage de données GPS afin d'explorer les chemins qu'empruntent les camions lourds dans le réseau routier urbain de Montréal. L'emphase est mise sur les voyages qui interagissent avec des terminaux intermodaux (gare de triage, port). Nous démontrons que les déplacements des camions peuvent être représentés avec précision et nous proposons un modèle logit récursif pour choix de chemin basé sur ces données. Ceci nous donne une meilleure compréhension des principaux facteurs qui influencent les décisions de déplacement et peut potentiellement nous permettre de minimiser les impacts des mouvements des camions, comme les émissions de CO2.
Mots clés : camions lourds, grand réseau urbain, données GPS, choix de route, logit récursif, terminaux intermodaux, cas d'étude

ABSTRACT. We explore the use of GPS devices to capture heavy truck routes in the Montreal urban road network. We emphasize on trips that interact with intermodal terminals (rail yard, port). We show that truck movements can be accurately represented and we propose a recursive logit model for route choice based on this data. This provides a better understanding in the main factors affecting the movement decisions and could potentially offer opportunities to reduce impact on truck movements, such as CO2 emissions.
Keywords: heavy trucks, large urban network, GPS data, route choice, recursive logit, intermodal terminals, case study

1. Introduction

Moving freight in urban areas is crucial for society and for economic growth. This chapter focuses on the analysis of heavy truck movements between intermodal terminals situated in an urban area. Major intermodal terminals generate a significant amount of traffic and have an important economic value. Ensuring efficient transport to and from these terminals is crucial, not only for the performance of the terminal operations and the trucking companies, but also for reducing the negative impacts of truck traffic, e.g., congestion, noise and emissions.

We explore a dataset of GPS traces from multiple large trucking companies active in the Greater Montreal area. These companies are members of an organization called CargoM - Logistic and Transportation Metropolitan Cluster of Montreal. CargoM was in charge of the data collection over three months during the winter of 2013-2014 using 48 GPS loggers.

While mobility patterns of people in urban areas have been extensively studied using GPS data, the literature on its freight counterpart is scarce. Most of the studies analyzing or modeling truck routes using GPS data focus on intercity transport and only a few investigate this problem in urban areas. Moreover, the studies are faced with important data processing challenges. In this study we illustrate through a case study how to generate results useful to stakeholders with only limited data processing. For this purpose, we report both descriptive results and results from structural modeling, in our case, random utility discrete choice models. We use the state-of-the-art methodologies and provide empirical contributions.

The objective of this study is first a descriptive analysis describing heavy truck movements on the island of Montreal and their access times to the two main intermodal terminals (a rail yard in the west and the Port of Montreal in the east) as well as the CO2 emissions on these routes. A second objective is to estimate a route choice model based on map-matched trajectory data. Data processing is typically a time-consuming and error-prone step prior to route choice modeling. The objective of this study is to keep this step to a minimum. Therefore, the descriptive study is based on raw GPS traces where only clearly erroneous data records have been removed. Moreover, the route choice model is link-based and only requires a network representation and map-matched trajectory data, unlike the more commonly used path-based models that require path choice set generation. While the size of the dataset is fairly limited, the main contribution of this study is to illustrate the feasibility of this approach on a large network.

This article is structured as follows. First, we present a brief literature review covering studies of freight movements by trucks using GPS data and discuss route choice modeling in this context. We then present the data followed by the results section. Finally, a conclusion summarizes our findings.

2. Literature Review

Multiple studies have shown the usefulness of GPS data in the analysis of truck movement patterns. GPS data provide a passive way of collecting trajectory data that overcomes the limits of fixed-date freight surveys, knowing that truck movements can vary considerably within weekdays [63]. [52] explores the use of smartphone applications for GPS tracking in order to gather data. GPS data can also be used to complement surveys to provide more information [66, 1, 75]. Truck movements are also subject to heterogeneity in the travel patterns which may be captured by GPS [7].

The main challenges of GPS data analysis often rely on the identification of tours and stops from the GPS traces, as well as information on the transported goods including empty trips. A number of studies [29, 65, 61, 68, 63] have adressed the issue by using various pre-processing methods to remove insignificant trips and by setting thresholds on time spent idle in order to identify a stop. [72] even propose a method to identify tour stops in urban areas with a machine learning algorithm (Support Vector Machine). In our study, the data includes a trip idenfitication number (identified by the driver or by an engine on-off event) which addresses the problem of finding stops within a continuous stream of GPS traces.

[22] propose a method to calculate freight performance measures in corridors. [31] analyse the efficiency of truck rail intermodal connectors using travel times for different periods of the week. [44] look at the speed associated to each GPS trace to analyze freight vehicle movements around São Paulo. They also analyze the data to identify stops made for deliveries.

While there are many route choice modelling studies focusing on car or public transport choices of individuals, there are relatively few studies on truck route choices. [32] estimate path size and error component models for heavy trucks intercity route choices using the previously described approach using generated choice sets and they provide an analysis of forecasting results. [67] analyze attributes that influence truck routing using a combination of two data sources: interviews at highway truck stops and stated preference data. [35] estimates various decision factors for urban commercial vehicle movements, in particular in the context of vehicle tours.

Most route choice models in the literature are random utility discrete choice models. We focus on this type of model and maximum likelihood estimation of model parameters using revealed preference data (trajectories in real networks). Route choice models are central in many transport applications. For example, the analysis of parameters allows to assess drivers' preferences towards different types of infrastructure or sensitivity to tolls. Moreover, the estimated models can be used to predict traffic flows. The literature on discrete choice models for route choice analysis can be grouped into three categories: (i) path-based models with generated choice sets of paths that are treated as actual choice sets ([58] provides an overview), (ii) path-based models based on universal choice set (all paths) but where choice sets are sampled and utilities corrected for the sampling [25] (iii) link-based recursive models that are based on universal choice without any choice sets of paths [47]. The third has major advantages over the first two because the model can be consistently estimated without the time-consuming process of choice set sampling and it can be used to compute predicted traffic flows in short computational times [26]. It, however, requires link-additive utilities. In this study, we estimate a recursive logit model [24].

This study uses GPS data to construct a route choice model for trucks in the context of intermodal trips in the Montreal transportation network, where two major logistics centres are present: the port of Montreal and a rail terminal. In doing so, we aim to provide a methodology for not only descriptive GPS data analysis but also route choice model estimation in an urban setting.

3. Data and Methodology

In this section, we start by giving a brief overview of the data collection process and the resulting dataset. We then discuss the map-matching procedure followed by a brief description of the recursive logit model.

3.1. Data

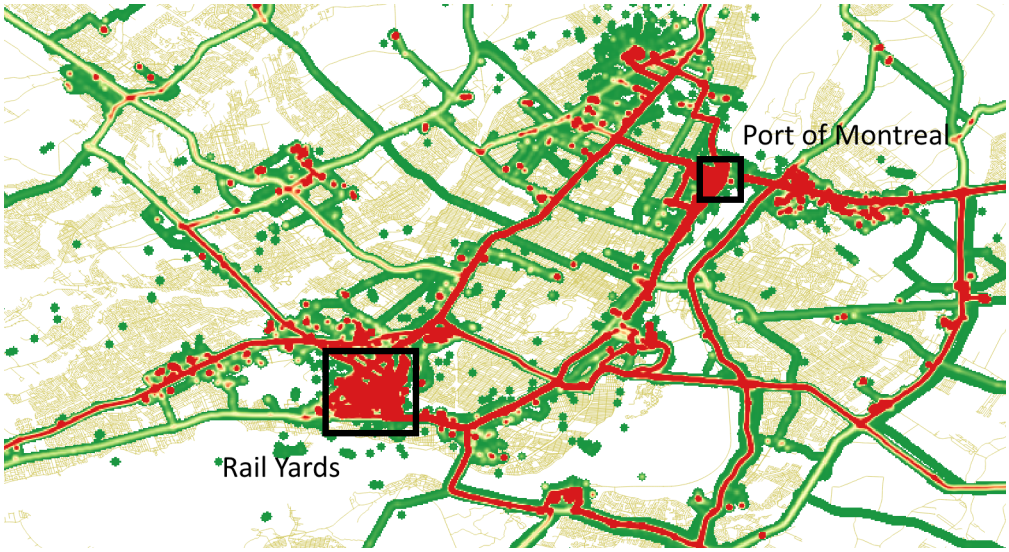
The data used in this study was collected on heavy trucks operating during a period of roughly 3 months, between December 9, 2013, and March 4, 2014, in the Greater Montreal

Area (GMA). These vehicles, owned by large trucking companies, primarily moved between the following locations: a major intermodal rail terminal in the center of Montreal Island, intermodal container terminals of the port of Montreal and various warehouses. A data point was recorded for every second when the vehicles' engine was on. The complete dataset contains 41,569,050 records corresponding to a total of 21,681 trips collected by 48 different vehicles. This gives us an average trip duration of 31.95 minutes which is reasonable given a typical trip length and the road network. Only 0.27% of trip records had invalid values. It is important to note that a new trip is started whenever the engine is switched off or if the driver specifies it through the collecting unit's interface. This means that many of these so-called trips are not significant both in distance traveled and in number of links used. This issue is dealt with during the map matching process that we discuss in the following section.

Figure 1 displays a heat map presenting the density of the data, red indicates a high density of data points while green represents a lower density. Highlighted are the port of Montreal and the intermodal rail yards. Two highways link the two locations and attract significant traffic. Three bridges and a tunnel are used to cross the Saint-Lawrence river running south of the island of Montreal.

Table 1 describes the contents of the GPS dataset, stored in a PostgreSQL database. Apart from the time and geographical information, the dataset also contains information on the cargo weight, provided by the truck driver via a small on-board interface.

Fig. 1. Heat map of the collected data



3.2. Map Matching

In order to estimate a route choice model, we need a description of the road network along with the observed route choices mapped to that network. In our case, all the information

Field	Description	Example values
Survey ID	Identification number of the data recording device.	496, 497, 498
Trip ID	Identification number of the current trip.	1, 2, 3, 4, 5, 6, 7, 8, 9, 10
Local Datetime	Date and time with precision to the second.	January 10, 2014, 09:24:01
Latitude	Latitude in degrees.	45.508928, 45.641103
Longitude	Longitude in degrees.	-73.509404, -73.762777
Heading	Heading in degrees.	78, 186, 278
Wheel based Speed	Current vehicle speed (m/s).	10.56, 3.98, 16.13
Cargo Weight	Weight of the cargo being transported currently in various units (when specified by the driver).	0, 20000, 1556
Reason for Stop	If specified, number indicating the reason for a stop or the end of the trip.	1, 2, 3, 4, 5, 6, 7, 8
CO2 emissions	Instantaneous CO2 emission in g/s.	2.01, 4.37

Table 1. Description of the fields in the dataset

related to the network comes from Adresses Québec (http://adressesquebec.gouv.qc.ca). A Python script is used to extract the relevant information and format it so it can be read in Matlab which is the software we use for route choice model estimation. Observed paths are obtained through a simple map matching algorithm using the data described previously and the road network. While we devised the following simple algorithm taking into account the precision of our GPS records as well as the network data, other algorithms could have been used [59]. First, every record for a single trip is extracted, then a PostgreSQL query assigns a link to each record based on proximity, heading and the direction of the link. The associated link must be within 10 meters of the corresponding GPS traces and their headings can be, at most, 40 degrees apart. If multiple links meet those criteria, then the closest one is retained. Afterwards, this list of links is iterated through by another Python script to build an observed path. At this stage, links must be sequential and certain thresholds are in place to filter out incorrectly matched links. To find an initial link, at least 5 GPS traces must be matched to the same link. Subsequently, a new link is added only when at least 3 consecutive GPS traces match to it. This process is then repeated for each trip. In order to estimate the

recursive logit model, we only keep the 698 observed paths that have a minimum of 6 links because longer trips provide more information.

3.3. Recursive Logit Model

Traditional route choice models represent the alternatives offered to the decision maker as the paths in the network, associating to each path a utility. It is assumed that the drivers aim to maximize their utility, and typically, a logit model is calibrated over the observed path choices. This modeling has, however, two major issues as the feasible paths in a real network cannot be enumerated and we do not know the set of paths actually considered by the decision maker. In addition to potentially long computational times, the obtained estimates can be biased and the predictions inaccurate. The recursive logit model [24] allows to circumvent these issues by assuming that the choice of the path is a sequence of link choices. A link is defined by its source and sink nodes in the network. At each choice stage, the driver chooses from the outgoing links the one that maximizes the sum of the instantaneous utility and the expected maximum utility until the destination (so-called value function). The instantaneous utility of link $a \in A(k)$ is $u(a|k;\beta) = v(a|k;\beta) + \varepsilon(a)$, where $A(k)$ is the set of outgoing links from link k, $v(a|k;\beta)$ is the deterministic utility and β is a vector of parameters to be estimated. The $\varepsilon(a)$ are independently and identically distributed extreme value type I (with location zero and scale μ) so that the choice model at each choice stage is logit. In this context, the value functions are given by the well-known logsum formula and Fosgerau et al. [24] show that they can be computed by solving a system of linear equations. Moreover, the probability of an observed path is the product of the link choice probabilities. The recursive logit model can be estimated and used for prediction without sampling any choice sets of paths as long as path utilities are link-additive and that the path utilities are given as the sum of the deterministic utilities of all links composing the path. We also note that an attribute similar to path size (see [6] for details) can be added to the utilities to correct for correlations. This attribute is called link size. We refer the reader to [24] and the tutorial [77] for more details on the model and its calibration.

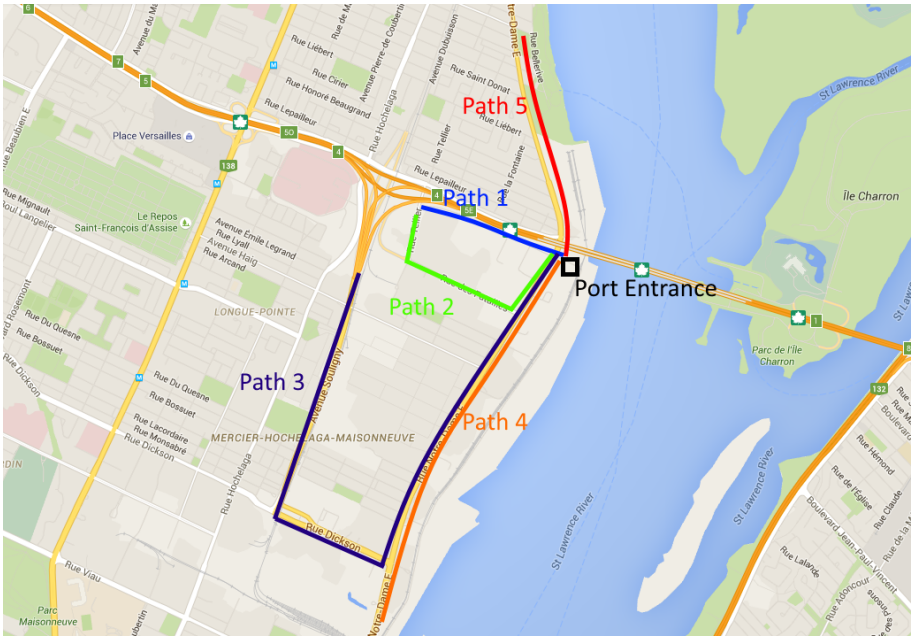
4. Results

In this section, we first present an illustrative example of a descriptive analysis of the observed path choices. The scope of the descriptive analysis of the case study was identified by the stakeholders. The descriptive anlysis paved the way to the development of the route choice model, whose results are presented hereafter.

4.1. Descriptive Analysis: Illutrative Example

The paths and associated travel times for trips with an intermodal terminal as origin or destination are important as they have an impact on the performance of the trucking operations, in particular, waiting times at the access points. These paths are also of importance for planning the network to avoid queues building up. As an example, Figure 2 illustrates the different street paths that can be used to access the "de Boucherville" entrance of the Port of Montreal.

Fig. 2. Travel paths to access "de Boucherville" port entrance



In our data, we observed a total of 1,131 entries made to the port at this location. The most used paths are path 1 (60.7% of occurrences) and path 4 (11.4%) while the other three paths are not significantly used (a combined 3.8%). The remaining 24.1% of entries could not be matched to any of the five paths. This is explained by the fact that these vehicles stopped at the warehouses in the area between paths 1 and 2 for some time before making a technically new trip to the port entrance. Since path 1 is the only path with a significant number of observations, we present a more detailed analysis of its travel times in Figure 3. The travel times slowly increase during the morning. More variability is observed from 10:00 to 13:00, a "peak" period for the port activities. This is reflected in the relatively high number of outliers and in the top whisker (the "maximum"). The dotted line indicates the travel time while driving at the legal speed limit. A similar analysis has been done for the exits from the port. It has contributed in enhancing traffic management at the "de Boucherville" exit: traffic signal timings have been changed to improve the situation.

Fig. 3. Travel time variability to access entrance using path 1, by time of the day

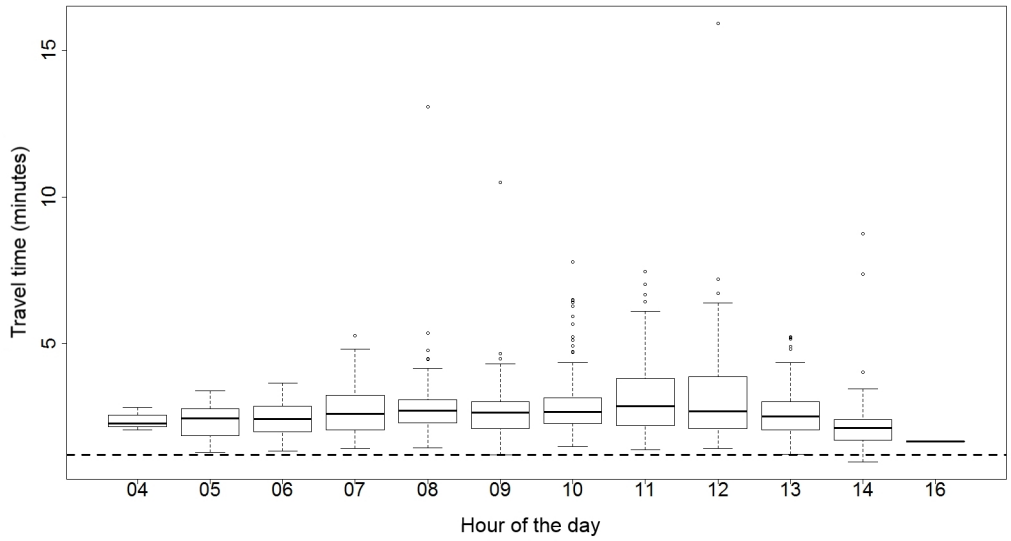
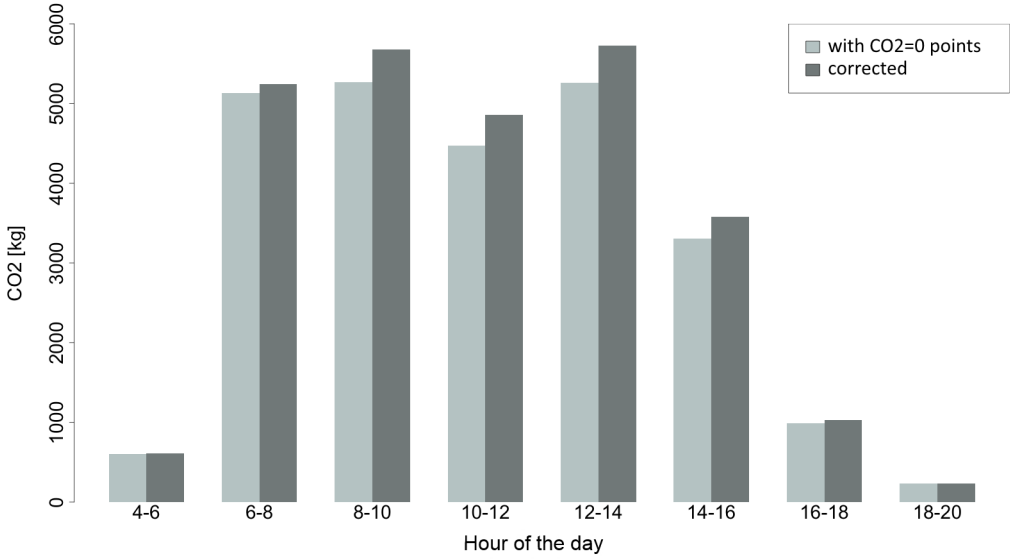


Fig. 4. CO2 emissions in the Port of Montreal



We also conduct an analysis of CO2 emissions in the Port of Montreal as a function of the time of day. First, we note that a small percentage of observations have a missing instantaneous CO2 emission value despite registering a movement speed above 0 m/s. There are different ways to impute these missing values. In our case, there are relatively few missing values so we use a crude method where we simply impute an average value from the observations, 2.05 g/s. Figure 4 reports the observed average emissions in kilograms over the time of day. We separate the results for observations with and without imputed values.

Furthermore, 4,836,698 records, representing 60.2% of the 8,034,382 records that are located within the area of the port, have a movement speed registered at 0 m/s. This means that the vehicle is not moving but has the engine on (idling). Summing the emissions for these observations, we have 8.9 tons of CO2 from stationary vehicles over the course of the 3 months during which the data was collected. Of course, for this number to have any meaning, we must calculate the percentage of the total number of trucks entering the port our sample represents. Using the number of trucks entering the port per day in our data and the real number recorded by port authorities, we estimate that our dataset represents 1.46% of all traffic. If we extrapolate using these numbers, idling vehicles emitted up to 550 tons of CO2 during the 3-month period. This brings us to develop a path choice model to, in the end, better understand the behavior of the drivers. These insights and route choice predictions can be used to improve the infrastructure to reduce the emissions.

4.2. Recursive Model Estimation

In this section, we present the estimation results of the recursive logit model without link size. The model is based on a large network containing 215,403 links and 115,157 nodes covering the whole region of the observations. The utilities are linear-in-parameters and are a function of the three following attributes: travel time (TT), link constant (LC) and left turns (LT). We recall that these utilities are link-additive. The deterministic part of the utility of an arc a is defined as $v(a|k;\beta) = \beta_{\mathrm{TT}} TT(a) + \beta_{\mathrm{LT}} LT(a|k) + \beta_{\mathrm{LC}} LC(a)$. So, the path travel time is the sum of the link travel times, left turns correspond to the number of left turns in a path and the path interpretation of link constant is the number of crossings. Based on statistical testing (t-test and likelihood ratio test), we analyzed different attributes and model specifications. Here, we report the model with the best in-sample fit.

We use an open-source Matlab code (https://github.com/maitien86) to estimate the model by maximum likelihood. It is an implementation of the nested fixed point algorithm proposed by [39], used in [47]. We can estimate the recursive logit model in a relatively short computational time using the decomposition method developed in [46]. There is an additional time required to load the data in Matlab, however, this loading time is also short (the computations were done on a machine with an i7 processor and 12 GB of RAM running Debian 3.16; the loading time is approximately 30 minutes and the run time is around 20 minutes). The computational time for the link-size model is longer because the utilities become origin-destination specific and the decomposition method cannot be used.

The estimation results are reported in Table 2. We note that the parameter estimates have their expected signs (negative) and they are significantly different from zero. This implies that an increase in expected travel time and left turns decreases the overall utility. The relatively high magnitude of the travel time parameter can be explained by the fact that the unit of time is in hours. Therefore, expected travel time values in hours for any link

would be quite small and, when multiplied by the travel time parameter, would not simply overtake every other parameter. In fact, we can interpret the ratios of the parameters relative to each other: $0.88/\ 353.52 = 0.002$. This means that a left turn is equivalent to adding 0.002 hours, or 8.95 seconds, to the travel time in our average truck driver's perception of route choice. Applying the same logic to the link constant parameter, which represents crossings, we have that a single crossing is equivalent to 6.24 seconds of additional travel time.

The purpose of our case study was to analyze in-sample results and, in that regard, the results are satisfactory. We note that the estimated model can be used to simulate truck route choices between different OD pairs and to compute traffic flows using an OD matrix as input in short computational times similar to [78].

Parameters	Recursive logit without link size	Std Error	t-test
Travel time value	-353.52	41.90	8.44
Left Turn	-0.88	0.092	9.56
Link Constant	-0.61	0.05	13.47

Table 2. Route choice model estimation results

5. Conclusion and Future Work

In this chapter, we analyzed truck drivers route choice behavior in two ways. First through a descriptive analysis based on raw GPS data and second, through the estimation of a recursive logit model. While the dataset is too small to draw general conclusions for the population of truck drivers in the Montreal region, the results illustrated that insights can be gained with a limited data processing effort. This holds true in particular when the interest lies in analyzing a specific region of the network, in our case the entry paths to the Port of Montreal. We also illustrated that the recursive logit model can be applied in a very large network. The model has the advantage of not requiring any generation of path choice sets, which is typically a time-consuming and error-prone part of route choice modeling. Future research should be dedicated to the analysis of truck tours, an aspect that we have ignored in this study since the focus was set on trips with one of the intermodal terminals as origin or destination.

Acknowledgements

The authors wish to acknowledge the funding and the support of CargoM, the logistics and transportation cluster of Montreal, for providing the data used in this study. We are grateful to Tien Mai for helping us with his MATLAB code used for the recursive logit model estimation. This research was partially funded by the National Sciences and Engineering Research Council of Canada, discovery grant 435678-2013.

Second Article.

Flow Capture under Heterogeneous User Behavior in Uncongested Networks

by

Léonard Ryo Morin[1], Emma Frejinger[1], and Bernard Gendron[1]

([1]) Department of Computer Science and Operations Research and CIRRELT
Université de Montréal, Canada

This article will be submitted to Operations Research before the thesis defense.

My contributions for this paper include implementing and refining the various models, devising the numerical experiments, producing the results and writing the text. Emma Frejinger helped detailing the theory around discrete choice. Bernard Gendron provided the initial mathematical developments leading to the Benders reformulation. Both Emma Frejinger and Bernard Gendron helped rewrite and reorganize certain sections.

Résumé. Nous considérons le problème de capture de flot dans lequel un gérant de réseau de transport décide sur quels arcs installer des ressources qui interceptent le trafic tout en affectant leurs chemins. Dans l'état de l'art de ce problème, les flots de trafic sont déterminés par des hypothèses simplistes comme des utilités déterministes et des ensembles de choix. Notre première contribution est une formulation qui incorpore un modèle stochastique d'affectation de trafic basé sur les arcs qui présume que les usagers font des choix maximisant des utilités aléatoires. Ceci permet de prendre en compte une variété de préférences incluant différentes perceptions des resources (positive, indifférente, négative). Pour obtenir cette formulation, nous proposons un programme biniveau à partir duquel nous dérivons plusieurs formulations dites "mixed integer linear". À partir d'une d'entre elles, obtenue en utilisant la dualité forte de manière particulière, nous dérivons une nouvelle méthode de décomposition de Benders. Notre seconde contribution est cette décomposition. Finalement, nous menons des expériences numériques sur un grand réseau afin de démontrer que des instances relativement grandes peuvent être résolues en un temps raisonnable.
Mots clés : Problème de capture de flot, comportement d'usagers, choix de chemin stochastique, logit récursif, décomposition de Benders

Abstract. We consider the general flow capture problem where a transportation network manager decides on which arcs to locate traffic intercepting resources that affects traffic flows. In the current state of the art surrounding this problem, traffic flows are determined under simplistic hypotheses such as deterministic utilities and path choice sets. Our first contribution is a formulation which integrates an arc-based stochastic traffic assignment model based on the assumption that travellers make random utility maximization choices. This allows to account for a variety of preferences, including different perceptions (positive, indifferent or negative) of the resources. To reach this formulation, we propose a bilevel program from which we derive several general mixed integer linear program formulations. From one of them, which is obtained through an uncommon use of strong duality, we derive a novel Benders reformulation. Our second contribution consists of this particular decomposition method. Finally, numerical experiments conducted on a large network show that relatively large instances can be solved in reasonable times.
Keywords: Flow capture problem, user behavior, stochastic path choice, recursive logit, Benders decomposition

1. Introduction

Flow capture problems (FCPs) concern the decisions on where to locate resources in a network so as to capture traffic flows. They can be viewed as facility location problems with the distinguishing feature that demand is represented by traffic flows as opposed to being static in specific locations. The objective of FCPs, subject to various constraints (e.g., budget), can be expressed, for example, as maximizing the amount of captured flow or as minimizing the consequences of non-captured flows. FCPs form a broad class of problems and encompass a variety of applications such as optimal location of rail park-and-ride facilities [34, 41], vehicle inspection stations [27, 55] and alternative fuel stations [60, 45, 42, 71].

[4] propose to categorize FCPs into three classes based on the assumption on the traffic flows in regards to the presence of resources. We refer to these classes as *indifferent*, *cooperative* or *evasive* flows. Cooperative and evasive flows clearly depend on the resource location decisions as the former seek them out, while the latter avoid them. These traffic flows are a result of travellers' route choice behavior where they seek to minimize their individual generalized cost function. This is known as the traffic assignment problem: given an origin-destination (OD) matrix containing the number of trips between each origin and destination in the network, and given a route choice model, the traffic flows are computed assuming that the travel time is either a function of the traffic volume or is constant (i.e., no congestion). In this work, we take the latter point of view, which is consistent with the state of the art on FCPs (see Section 2).

Our contribution, however, significantly extends the state of the art, as the FCP models proposed in the literature are based on the assumption that travellers behave in a known and deterministic way, that is, travellers minimize a perfectly known generalized cost function so the route choice problem is reduced to a simple shortest path problem. There is a large gap between this strong assumption and the state of the art in route choice models. In fact, there is ample empirical evidence in the literature focused on route choice analysis [77] that travellers' generalized cost functions cannot be known perfectly. In turn, this motivates the use of *random utility maximization* (RUM) models, which have better prediction accuracy. Moreover, the spatial overlap of paths in a transportation network requires RUM models that allow utilities to be correlated. The simplest RUM model – logit, aka logistic regression – is based on the assumption that utilities are uncorrelated and has a poor prediction performance compared to models that allow for correlated utilities. Most network optimization formulations that integrate user behavior use a logit model (see Section 2). Note that, even this simplest setting leads to non-linear non-convex formulations.

We aim to fill the gap between the state of the art on FCPs and route choice models. This paper offers four main contributions:

(1) We propose a bilevel programming model for the FCP where the lower-level route choice decisions are given by an arc-based RUM model, called nested recursive logit [47]. It allows to predict traffic flows without any enumeration of choice sets of paths – i.e., the support for the probability distribution – and path utilities that are correlated. By appropriately defining the utility functions, the model can predict traveller behavior according to any class of flows – evasive, cooperative or indifferent – and accounts for the heterogeneous behavior within each class.

(2) Based on the simulation approach proposed in [56], we express the RUM model in arc utility space instead of arc probability space. On the one hand, this leads to linear constraints instead of non-linear ones. On the other hand, the resulting model has

a large number of variables and constraints. This is particularly challenging in our case as, unlike [56], we focus on the more complex network optimization setting.

(3) The bilevel programming model is rewritten as a single-level model through the addition of a Lagrangian term derived from strong duality constraints as opposed to simply including them as constraints or by including complementary slackness constraints. The structure of the resulting model can be exploited to devise an efficient Benders decomposition that relies heavily on solving shortest path problems rather than linear programs, thus allowing large instances to be solved in reasonable times.

(4) We present results from extensive computational experiments on both a small illustrative network and the larger Winnipeg network, commonly used as a benchmark for evaluating traffic assignment models [e.g., 3, 57, 37, 38]. Experiments on the small network allows us to assess the impact of various problem characteristics and to test the scalability of the single-level models when solved with a state-of-the-art solver, thus motivating the use of Benders decomposition for the larger Winnipeg network.

The remainder of the paper is structured as follows. Section 2 contains a review of the relevant literature. Section 3 describes the problem and the bilevel programming formulation integrating the simulation approach. Section 4 details various techniques to achieve single-level formulations, which are then linearized in order to exploit state-of-the-art mixed-integer linear programming (MILP) solvers. Section 5 presents the Benders decomposition method. Section 6 focuses on our numerical experiments, while Section 6 concludes the paper and proposes avenues for future research.

2. Literature Review

This chapter is focused on bridging the gap between the state of the art in demand modeling – in this case route choice behavior – and network optimization for FCPs. There are two aspects capturing the demand in any FCP formulation. First, there is what we refer to as *aggregate* demand, the number of trips between each OD pair over a given period of time, captured in an OD matrix. Second, there is how these trips are distributed over paths in the network in response to the resource allocation decisions. We refer to this as *disaggregate* demand. Most studies on FCPs are single period, however [51] consider a multi-period problem. Aggregate demand is assumed to be fixed and given with a notable exception in [71] where aggregate demand is stochastic. Accordingly, we focus on a single period problem assuming deterministic aggregate demand. Our contribution focuses on integrating state of the art disaggregate demand models. In the following we provide an overview of disaggregate demand models in the context of FCPs. We then review the state of the art in RUM route choice models followed by a high-level description of bilevel programming models where the lower-level user response is given by a RUM model. Finally, we describe the simulation approach of [56] on which we base this work.

Disaggregate demand is incorporated into FCPs in various ways across the literature. In the seminal works of [33] and [8], the demand is assumed to travel along the predetermined shortest path for each OD pair. The concept of deviation, where users are willing to deviate from the shortest path to seek out or avoid a facility, is integrated in several subsequent works [e.g., 74, 42, 50]. This is mainly done through path set enumeration, which implies a limitation in regards to the number of paths considered. More recently, [4] improved on this by reworking the model proposed in [50] to dynamically find the shortest path with an arc-based formulation. However, their model remains deterministic similarly to the path enumeration approaches. To the best of our knowledge, there are no RUM models used to represent the disaggregate demand in FCPs, which is the gap we address in this work.

RUM models are frequently used to analyze and to predict route choice behavior in transportation networks. There are two key challenges in this context that have been extensively studied: (i) the definition of choice sets and (ii) modeling correlated utilities in a computationally tractable manner. Recursive models proposed by [24], [47] (aka maximum entropy inverse reinforcement learning) [76] address these challenges. We refer to [77] for a comprehensive overview of related work. Based on dynamic programming, these models predict path choices based on an arc formulation without any restrictive assumption about the choice sets, that is, any feasible path in the network is part of the choice set. We explore this nice property.

While RUM models have not been included in FCPs, they have been integrated in other problems. An example is the logit network pricing detailed in [28]. The solution approach consists of solving a linear approximation of the problem and then using a local search method. There are three possible approximations presented in their work. The first consists of using a deterministic path assignment. The second and third consist of replacing the non-linear term describing the choice probability of each alternative by a constant function and a linear function, respectively. [49] consider a facility location problem with a RUM (logit) model to calculate choice probabilities. A heuristic based on GRASP (greedy randomized adaptive search procedure) and tabu search is proposed. Using [49] as a starting point, [20] reformulate the model as a bilevel program with endogenous facility service rate variables amongst its improvements. Their solution method is a heuristic based on that of [28].

An alternate method of incorporating a RUM model, the one we use in our work, is based on the simulation approach of [56]. They propose to integrate a sample average approximation of predictions from discrete choice models into MILP formulations. Instead of working with the non-linear expressions representing choice probabilities, the approach focuses on drawing random terms from the distribution of the choice model to obtain utility values. Following a principle of utility maximization, user choices correspond to the alternatives having maximum utility. This approach, to our knowledge, has not yet been used in

a network optimization context, which exhibits specific challenges compared to the setting considered by [56].

In summary, the gap we address in this paper is the lack of RUM models in the FCP literature. This is achieved by adapting a simulation approach to a network optimization setting, which is also a new development. The resulting bilevel programming formulation is presented in the following section.

3. Problem Description and Bilevel Model

In this section, we introduce the FCP that we consider in our work, along with a bilevel programming model. In this context, the decision maker chooses where to locate different types of resources $r \in R$ on the arcs of an uncongested network $G = (N,A)$ composed of nodes $n \in N$ and arcs $a \in A$. The resources available on each arc $a \in A$ are identified with the indicator parameter σ_{ar} that takes value 1, if resource $r \in R$ can be installed on arc $a \in A$, and value 0, otherwise. Note that we might have $\sum_{r \in R} \sigma_{ar} = 0$, in which case arc $a \in A$ cannot be used to locate any resources. The objective is to maximize the overall captured traffic flow. Associated with each arc $a \in A$ and each resource $r \in R$ is an installation cost $c_{ar} > 0$ and a proportion of the flow the resource can capture $q_{ar} \in (0,1]$. The decisions may be subject to different constraints, such as, not exceeding an overall budget $b > 0$, and imposing a maximum or a minimum number of resources to install on subsets of the arcs. For the sake of simplicity, we only impose the constraint that a single resource can be installed on each arc. The resource location decision variables y_{ar} are equal to 1, if a resource of type $r \in R$ is installed on arc $a \in A$, and to 0, otherwise. We then assume the constraints on the resource location variables to be captured by set

$$Y = \left\{ y \in \{0,1\}^{|A|\times|R|} \mid \sum_{a \in A} \sum_{r \in R} \sigma_{ar} c_{ar} y_{ar} \leq b; \sum_{r \in R} \sigma_{ar} y_{ar} \leq 1, a \in A \right\}.$$

Accordingly, in the remainder of the paper, we write these constraints in the compact form $y \in Y$.

Users observe the location of the resources and make route choices between different OD pairs $k \in K$ according to their preferences. For example, they can be attracted to the resources (henceforth cooperative users), indifferent to them, or they might want to avoid them (henceforth evasive users). They can also have different preferences regarding other characteristics of the network, such as distance, presence of traffic lights, specific road types and speed limits.

As the objectives of the decision maker and the users are not necessarily aligned, the problem can be formulated as a bilevel program where the leader is the decision maker and the followers are the users of the system. We divide the users into categories $l \in L$ depending on their preferences and type of behavior. Also, we assume that the aggregate demand is

fixed and known, meaning that we know the number of users d_l^k of each category $l \in L$ travelling between the origin $O(k)$ and the destination $D(k)$ of each $k \in K$.

Before presenting the formulation, we turn our attention to the model of user behavior. In order to predict the route choice behavior of the users in each category, we rely on additive RUM models. These models are based on the assumption that each user category $l \in L$ associates a *dis*utility $u'_{lp}(y,z,\varepsilon')$ with each alternative $p \in P$ and selects one of them (note that we describe the model in terms of minimizing a disutility instead of maximizing a utility, the latter being standard in the demand modeling literature). For the sake of notation brevity, we omit the model parameters from this expression. The disutility is assumed to depend on the resource location decision variables y, but also on the attributes of the network and of the users, captured in the exogeneous variables z, and on a random variable ε. For the sake of simplicity, we start with a *path-based model*, so p in this context is a path and P is the set of all paths in the network for all OD pairs, i.e., $P = \cup_{k \in K} P^k$, where P^k is the set of all paths for OD pair $k \in K$ (in the remainder, we assume that all paths are elementary). In order to define a model for the choice of a path, we make the following assumption on the associated disutility:

Assumption 1. *The disutility of path $p \in P$ for user category $l \in L$ is*

$$u'_{lp}(y,z,\varepsilon') = g'_{lp}(y;\beta'_l) + h'_{lp}(z;\alpha'_l) + \varepsilon'_{lp}, \tag{3.1}$$

where

- *$g'_{lp}(y;\beta'_l)$ is a function that depends on the resource location decisions $y \in Y$ and β'_l is a vector of preference parameters;*
- *$h'_{lp}(z;\alpha'_l)$ is a function of a vector z of attributes of the network and of the users (we assume z takes its values in some set Z), and α'_l is a vector of preference parameters;*
- *ε'_{lp} is a random variable that determines the type of RUM model through its distributional assumption.*

The probability that a user in category $l \in L$ chooses alternative $p \in P^k$ for OD pair $k \in K$ is equal to the probability that its disutility is less than the disutility of any other path: $\mathrm{P}(u'_{lp}(y,z,\varepsilon') < u'_{lp'}(y,z,\varepsilon'), p' \neq p,\ p',p \in P^k)$. Depending on the assumption on the random variables, this probability may have a closed form. However, even for the simplest RUM model (logit) – based on the assumption that ε'_{lp} are i.i.d. extreme value type I distributed – this probability is non linear in y, even if $g'_{lp}(y;\beta'_l)$ is linear.

Instead of probabilities, we write our formulation in the space of disutilities. More precisely, a path-based model of our FCP makes use of path choice variables x'^k_{lp}, assuming value 1, if path $p \in P^k$ is selected by user category $l \in L$ to travel from $O(k)$ to $D(k)$, and value 0, otherwise. For each OD pair $k \in K$, set X'^k_l captures the constraint that each user category

$l \in L$ selects a single path:

$$X_l'^k = \left\{ x_l'^k \in \{0,1\}^{|P^k|} \mid \sum_{p \in P^k} x_{pl}'^k = 1 \right\}.$$

The bilevel programming model of our FCP can then be written as follows:

$$\max \sum_{a \in A} \sum_{r \in R} \sigma_{ar} q_{ar} y_{ar} \left(\sum_{k \in K} \sum_{p \in P^k} \sum_{l \in L} \delta_a^p d_l^k x_{lp}'^k \right) \tag{3.2}$$

$$y \in Y \tag{3.3}$$

$$x_l'^k \in \arg \min_{x_l'^k \in X_l'^k} \{ \mathbb{E}_{\varepsilon'} [u_l'(y,z,\varepsilon') x_l'^k] \},\ k \in K, l \in L, \tag{3.4}$$

where δ_a^p is equal to 1, if arc $a \in A$ belongs to path $p \in P$, and to 0, otherwise. The objective of the leader, (3.2), is to maximize the flow capture subject to the resource location constraints (3.3) and to the constraints that each user category $l \in L$ chooses the path for each OD pair $k \in K$ that minimizes its expected disutility, (3.4).

One of the main issues with this formulation is related to the path sets $P^k,\ k \in K$, as they cannot be enumerated even for medium size networks. We therefore propose to write the formulation in arc flows, hence removing the dependence on paths. The reformulation relies on two key elements: properties of the recursive route choice models [24, 47] and a simulation approach (in the space of disutilities) to evaluate the expected arc flows, (3.4), through sample average approximation.

As opposed to path-based route choice models that require P as support for the probability distribution, recursive models decompose the choice of path in a sequential arc choice process (Markov decision process). They are based on the full network and path choice probabilities are computed by multiplying the probability of each arc in the path. This is possible assuming that path utilities are arc additive:

Assumption 2. *The disutility of path $p \in P$ for user category $l \in L$ can be expressed per arc as*

$$u_{lp}'(y,z,\varepsilon') = \sum_{a \in A} \delta_a^p u_{la}(y,z,\varepsilon) \tag{3.5}$$

where $u_{la}(y,z,\varepsilon) = g_{la}(y;\beta_l) + h_{la}(z;\alpha_l) + \varepsilon_{la}$ with similar interpretations as in Assumption 1, i.e., $g_{la}(y;\beta_l)$ is a function of $y \in Y$ and β_l is a vector of preference parameters; $h_{la}(z;\alpha_l)$ is a function of $z \in Z$ and α_l is a vector of preference parameters; ε_{la} is a random variable that determines the type of RUM model.

We can now propose an arc-based formulation of our FCP, where arc flow variables x_{la}^k assume value 1, if arc $a \in A$ is selected by user category $l \in L$ as part of its chosen path for OD pair $k \in K$:

$$\max \sum_{a \in A} \sum_{r \in R} \sigma_{ar} q_{ar} y_{ar} \left(\sum_{k \in K} \sum_{l \in L} d_l^k x_{la}^k \right) \tag{3.6}$$

$$y \in Y \tag{3.7}$$

$$x_l^k \in \arg\min_{x_l^k \in X_l^k} \{\mathbb{E}_\varepsilon[u_l(y,z,\varepsilon)x_l^k]\},\ k \in K, l \in L, \tag{3.8}$$

where X_l^k is the set of feasible arc flows that define all elementary paths for each $k \in K$ and each $l \in L$, i.e.,

$$X_l^k = \left\{ x_l^k \in \{0,1\}^{|A|} \mid \sum_{a \in F(n)} x_{la}^k - \sum_{a \in B(n)} x_{la}^k = e_n^k,\ n \in N\,;\ \sum_{a \in B(n)} x_{la}^k \leq 1,\ n \in N \right\},$$

where, for each node $n \in N$, we use the notation $F(n) = \{a \in A | a = (n,m)\}$ $B(n) = \{a \in A | a = (m,n)\}$, and

$$e_n^k = \begin{cases} 1, & \text{if } n = O(k), \\ -1, & \text{if } n = D(k), \\ 0, & \text{otherwise.} \end{cases}$$

The arc-based model, (3.6)-(3.8), is equivalent to the path-based one, (3.2)-(3.4). Indeed, on the one hand, any path flows can be written as arc flows that capture the same disutilities, thanks to Assumption 2. On the other hand, any arc flows that satisfy constraints (3.8) correspond to path flows that minimize the same expected disutilities, due to Assumption 2.

Before presenting the formulation obtained through sample average approximation, we explicitly define the expression for the disutility of an arc. We begin with the following assumption on $g_{la}(y;\beta_l)$:

Assumption 3. *For any arc $a \in A$ and any user category $l \in L$, the function $g_{la}(y;\beta_l)$ is linear and can be written as follows:*

$$g_{la}(y;\beta_l) = \sum_{r \in R} \beta_{lar}\sigma_{ar}y_{ar}. \tag{3.9}$$

Typically, $h_{la}(z;\alpha_l)$ is also linear, although this is not required for what follows, unlike Assumption 3.

Assumption 3 provides us with a simple characterization of user behavior. With respect to the installation of a resource of type $r \in R$ on arc $a \in A$ such that $\sigma_{ar} = 1$, a user category $l \in L$ is: cooperative, if $\beta_{lar} < 0$; indifferent, if $\beta_{lar} = 0$; evasive, if $\beta_{lar} > 0$.

Although this is not required in most of our developments, it is often convenient to assume that all users in a given category are either cooperative, or indifferent, or evasive (*homogeneity assumption*), where users in category $l \in L$ are:

- *cooperative*, if $\beta_{lar} \leq 0$ for each $a \in A$ and $r \in R$ such that $\sigma_{ar} = 1$, and $\sum_{a \in A}\sum_{r \in R} \beta_{lar}\sigma_{ar} \neq 0$;
- *indifferent*, if $\beta_{lar} = 0$ for each $a \in A$ and $r \in R$ such that $\sigma_{ar} = 1$;
- *evasive*, if $\beta_{lar} \geq 0$ for each $a \in A$ and $r \in R$ such that $\sigma_{ar} = 1$, and $\sum_{a \in A}\sum_{r \in R} \beta_{lar}\sigma_{ar} \neq 0$.

In some cases (see Section 5.4), we make the *strong homogeneity assumption*, which states that all users in a given category are either strongly cooperative or strongly evasive, where users in category $l \in L$ are:

- *strongly cooperative*, if $\beta_{lar} < 0$ for each $a \in A$ and $r \in R$ such that $\sigma_{ar} = 1$;
- *strongly evasive*, if $\beta_{lar} > 0$ for each $a \in A$ and $r \in R$ such that $\sigma_{ar} = 1$.

While the expected minimum disutility (3.8) is the solution to stochastic shortest path problems, computing a sample average approximation through simulation corresponds to solving deterministic shortest path problems over realizations of the arc disutilities. We explore such shortest path computations by using the simulation approach of [56], discussed in Section 2. We call one realization of ε a scenario $s \in S$ and denote the corresponding realization of arc disutilities $u^s_{la}(y,z,\varepsilon^s)$. By drawing a sufficiently large number of scenarios $|S|$, assigning traffic to the resulting shortest paths x^{ks}_l, $s \in S, k \in K, l \in L$, constitutes a sample average approximation of the stochastic flow distribution, i.e.,

$$x^k_l = \frac{1}{|S|} \sum_{s \in S} x^{ks}_l,\ k \in K, l \in L.$$

Note that, due to Assumption 3, the arc disutility for any scenario $s \in S$ can be written as:

$$u^s_{la}(y, \psi^s) = \sum_{r \in R} \beta_{lar} \sigma_{ar} y_{ar} + \psi^s_{la}, \tag{3.10}$$

where $\psi^s_{la} = h_{la}(z; \alpha_l) + \varepsilon^s_{la}$. For the sake of brevity, only this definition is inserted in subsequent mathematical developments when necessary.

We can now write a deterministic arc-based model that serves as a basis for subsequent developments:

$$Z = \max \sum_{a \in A} \sum_{r \in R} \sigma_{ar} q_{ar} y_{ar} \left(\frac{1}{|S|} \sum_{s \in S} \sum_{k \in K} \sum_{l \in L} d^k_l x^{ks}_{la} \right) \tag{3.11}$$

$$y \in Y \tag{3.12}$$

$$x^{ks}_l \in \arg \min_{x^{ks}_l \in X^{ks}_l} \{u^s_l(y,\psi^s) x^{ks}_l\},\ s \in S, k \in K, l \in L, \tag{3.13}$$

where X^{ks}_l is the set of feasible arc flows for $s \in S$, $k \in K$ and $l \in L$, i.e.,

$$X^{ks}_l = \left\{ x^{ks}_l \in \{0,1\}^{|A|} \mid \sum_{a \in F(n)} x^{ks}_{la} - \sum_{a \in B(n)} x^{ks}_{la} = e^k_n, n \in N\,;\ \sum_{a \in B(n)} x^{ks}_{la} \leq 1,\ n \in N \right\}.$$

This deterministic bilevel programming formulation has several interesting properties if the disutilities satisfy the following assumption.

Assumption 4. *For each arc $a \in A$, user category $l \in L$ and scenario $s \in S$, there exist constants μ and Δ such that*

$$\mu \geq u^s_{la}(y,\psi^s) \geq \Delta > 0,\ y \in Y. \tag{3.14}$$

The assumption $u^s_{la}(y,\psi^s) > 0$ easily holds in practice if the deterministic part of the arc disutility $g_{la}(y;\beta_l)+h_{la}(z;\alpha_l)$ is large enough (typically, the term $h_{la}(z;\alpha_l)$ includes the travel time on arc $a \in A$). The constants μ and Δ are used in our subsequent developments (see Section 4). Under Assumption 4, the disutilities are always positive and the set of feasible arc flows X^{ks}_l, for $s \in S$, $k \in K$ and $l \in L$, can now be written as

$$X^{ks}_l = \left\{ x^{ks}_l \in \{0,1\}^{|A|} \mid \sum_{a \in F(n)} x^{ks}_{la} - \sum_{a \in B(n)} x^{ks}_{la} = e^k_n, n \in N \right\}.$$

Shifting our attention back to (3.11)-(3.13), we see that for fixed $y \in Y$, the follower problem decomposes by scenario, user category and OD pair, and reduces to a shortest path problem, which can be solved by Dijkstra's algorithm, thanks to Assumption 4. Our Benders decomposition method, presented in Section 5, exploits this property. In addition, due to Assumption 3, the resource location and arc flow variables are only linked through bilinear terms, in the objectives of both the leader and the follower. We also make use of this property in our Benders decomposition method. Finally, for fixed $y \in Y$, the follower problem reduces to a linear program, since the incidence matrix of a directed graph is totally unimodular. This implies that we can relax the integrality constraints on the flow variables without losing optimality. This property is exploited to derive single-level reformulations that constitute essential steps towards the development of our Benders decomposition method. These reformulations are presented next.

4. Single-Level Reformulations

After relaxing the integrality constraints on the arc flow variables, we can write the follower problem for fixed $y \in Y$ and for scenario $s \in S$, OD pair $k \in K$ and user category $l \in L$ as a linear program:

$$\min \sum_{a \in A} u^s_{la}(y,\psi^s) x^{ks}_{la} \tag{4.1}$$

$$\sum_{a \in F(n)} x^{ks}_{la} - \sum_{a \in B(n)} x^{ks}_{la} = \begin{cases} 1, & \text{if } n = O(k), \\ -1, & \text{if } n = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad n \in N, \tag{4.2}$$

$$x^{ks}_{la} \geq 0, \quad a \in A. \tag{4.3}$$

Denoting as π^{ks}_{ln} the dual variables associated to constraints (4.2), the dual of this linear program can be written as follows:

$$\max \pi^{ks}_{lD(k)} \tag{4.4}$$

$$\pi^{ks}_{lm} - \pi^{ks}_{ln} \leq u^s_{la}(y,\psi^s), \quad a = (n,m) \in A, \tag{4.5}$$

$$\pi^{ks}_{lO(k)} = 0, \tag{4.6}$$

where we use the fact that (4.2) contains one redundant equation, which we associate to the origin $O(k)$. We can then replace the objective of the follower problem, (4.1), by the dual feasibility constraints, (4.5)-(4.6), and by optimality conditions, which can be the *complementary slackness conditions*

$$x_{la}^{ks}\left(u_{la}^{s}(y,\psi^{s})-(\pi_{lm}^{ks}-\pi_{ln}^{ks})\right)=0,\quad a=(n,m)\in A, \tag{4.7}$$

or the *strong duality constraint*

$$\pi_{lD(k)}^{ks}=\sum_{a\in A}u_{la}^{s}(y,\psi^{s})x_{la}^{ks}. \tag{4.8}$$

We can now write a single-level non-linear reformulation of (3.11)-(3.13) that combines both (4.7) and (4.8) (for the sake of completeness, we write the constraints in extensive form):

$$Z=\max\frac{1}{|S|}\sum_{s\in S}\sum_{a\in A}\sum_{r\in R}\sum_{k\in K}\sum_{l\in L}(\sigma_{ar}q_{ar}d_{l}^{k})y_{ar}x_{la}^{ks} \tag{4.9}$$

$$\sum_{a\in A}\sum_{r\in R}\sigma_{ar}c_{ar}y_{ar}\leq b, \tag{4.10}$$

$$\sum_{r\in R}\sigma_{ar}y_{ar}\leq 1,\quad a\in A, \tag{4.11}$$

$$y_{ar}\in\{0,1\},\quad a\in A,r\in R, \tag{4.12}$$

$$\sum_{a\in F(n)}x_{la}^{ks}-\sum_{a\in B(n)}x_{la}^{ks}=e_{n}^{k},\quad n\in N,s\in S,k\in K,l\in L, \tag{4.13}$$

$$x_{la}^{ks}\geq 0,\quad a\in A,s\in S,k\in K,l\in L, \tag{4.14}$$

$$\pi_{lm}^{ks}-\pi_{ln}^{ks}\leq u_{la}^{s}(y,\psi^{s}),\quad a=(n,m)\in A,s\in S,k\in K,l\in L, \tag{4.15}$$

$$\pi_{lO(k)}^{ks}=0,\quad s\in S,k\in K,l\in L, \tag{4.16}$$

$$x_{la}^{ks}\left(u_{la}^{s}(y,\psi^{s})-(\pi_{lm}^{ks}-\pi_{ln}^{ks})\right)=0,\quad a=(n,m)\in A,s\in S,k\in K,l\in L, \tag{4.17}$$

$$\pi_{lD(k)}^{ks}=\sum_{a\in A}u_{la}^{s}(y,\psi^{s})x_{la}^{ks},\quad s\in S,k\in K,l\in L. \tag{4.18}$$

4.1. Lagrangian Reformulation

Before linearizing the single-level formulation, we take a look at another method of guaranteeing optimality of the follower problem, based on penalizing the violation of the strong duality constraints (4.18). The key element here is to use Lagrange multipliers set to large enough finite values to guarantee the optimality of the follower problem. This approach is essential in order to derive our Benders decomposition method presented in Section 5.

We start by adding the strong duality constraints (4.18) as a Lagrangian term to the objective (4.9) using a vector of Lagrange multipliers $\lambda=(\lambda_{l}^{ks})_{s\in S,k\in K,l\in L}$. We thus derive

the following *Lagrangian-based single-level non-linear model*:

$$\begin{aligned} Z(\lambda) = &\max \frac{1}{|S|} \sum_{s\in S} \sum_{a\in A} \sum_{r\in R} \sum_{k\in K} \sum_{l\in L} (\sigma_{ar} q_{ar} d_l^k) y_{ar} x_{la}^{ks} \\ &+ \sum_{s\in S} \sum_{k\in K} \sum_{l\in L} \lambda_l^{ks} \left(\pi_{lD(k)}^{ks} - \sum_{a\in A} u_{la}^s(y,\psi^s) x_{la}^{ks} \right) \end{aligned} \tag{4.19}$$

subject to (4.10)-(4.18).

Proposition 1. *The Lagrangian-based single-level non-linear model satisfies the following properties:*

(1) Removing either (4.17) or (4.18) yields a reformulation: $Z(\lambda) = Z$ for any λ.

(2) Removing (4.17) and (4.18) yields a relaxation: $Z(\lambda) \geq Z$ for any λ.

(3) Removing (4.17) and (4.18), and setting λ to large enough finite values yields a reformulation: $Z(\lambda) = Z$ for any λ such that

$$\lambda_l^{ks} > \frac{d_l^k \sum_{a\in A} \max_{r\in R}\{\sigma_{ar} q_{ar}\}}{|S|\Delta}, \quad s \in S, k \in K, l \in L.$$

Proof. Since 1 and 2 are trivial, we only show 3. Let $(\overline{y},\overline{x},\overline{\pi})$ be an optimal solution of value Z to (4.9)-(4.18). Assume that we remove both (4.17) and (4.18) from the Lagrangian-based single-level non-linear model. Then, if we fix $y = \overline{y}$, the resulting subproblem decomposes by scenario $s \in S$, by OD pair $k \in K$ and by user category $l \in L$, and further decomposes into a subproblem in x variables only and a subproblem in π variables only. Thus, for $s \in S$, $k \in K$ and $l \in L$, the subproblem in x variables is

$$\max \sum_{a\in A} \left\{ \left(\frac{1}{|S|} \sum_{r\in R} d_l^k \sigma_{ar} q_{ar} \overline{y}_{ar} \right) - \lambda_l^{ks} u_{la}^s(\overline{y},\psi^s) \right\} x_{la}^{ks} \tag{4.20}$$

subject to (4.2)-(4.3), while the subproblem in π variables is

$$\max \lambda_l^{ks} \pi_{lD(k)}^{ks} \tag{4.21}$$

subject to (4.5)-(4.6). This last subproblem is the dual of a shortest path problem with arc lengths equal to $u_{la}^s(\overline{y},\psi^s)$. We denote as $\widetilde{\pi}_l^{ks}$ the solution to this problem. Concerning the subproblem in x variables, under the assumption on the values of λ, it holds that

$$\begin{aligned} \lambda_l^{ks} &> \frac{d_l^k \sum_{a'\in A} \max_{r\in R}\{\sigma_{a'r} q_{a'r}\}}{|S|\Delta} \\ &\geq \frac{d_l^k \sum_{a'\in A} \sum_{r\in R} \sigma_{a'r} q_{a'r} \overline{y}_{a'r}}{|S|\Delta}, \\ &\geq \frac{d_l^k \sum_{a'\in A} \sum_{r\in R} \sigma_{a'r} q_{a'r} \overline{y}_{a'r}}{|S| u_{la}^s(\overline{y},\psi^s)}, \; a \in A, \text{ (by Assumption 4)} \end{aligned}$$

which implies

$$\begin{aligned} t_{la}^{ks} &\equiv \lambda_l^{ks} u_{la}^s(\overline{y},\psi^s) - \frac{1}{|S|}\sum_{r\in R} d_l^k \sigma_{ar} q_{ar} \overline{y}_{ar},\ a \in A, \\ &\geq \lambda_l^{ks} u_{la}^s(\overline{y},\psi^s) - \frac{1}{|S|}\sum_{a'\in A}\sum_{r\in R} d_l^k \sigma_{a'r} q_{a'r} \overline{y}_{a'r},\ a \in A. \\ &> 0 \end{aligned}$$

Thus, the subproblem in x variables can be solved as a shortest path problem with arc lengths $t_{la}^{ks} > 0$. In addition, every single term $\lambda_l^{ks} u_{la}^s(\overline{y},\psi^s)$ is strictly larger than the global sum $\sum_{a'\in A}\sum_{r\in R} \frac{1}{|S|} d_l^k \sigma_{a'r} q_{a'r} \overline{y}_{a'r}$. This implies that the the solution $\widetilde{x}_l^{ks}$ to the shortest path problem with arc lengths t_{la}^{ks} is also optimal for the shortest path problem with arc lengths $u_{la}^s(\overline{y},\psi^s)$ (ties being broken with the second term $\frac{1}{|S|}\sum_{r\in R} d_l^k \sigma_{ar} q_{ar} \overline{y}_{ar}$, for any $a \in A$), which implies that

$$\widetilde{\pi}_{lD(k)}^{ks} = \sum_{a\in A} u_{la}^s(y,\psi^s)\widetilde{x}_{la}^{ks}.$$

Thus, we conclude that $Z(\lambda) = Z$ and $(\overline{y},\widetilde{x},\widetilde{\pi})$ is optimal for (4.9)-(4.18). □

The approach that consists of penalizing the strong duality constraint is well-known in the literature on bilevel programming (see, e.g., [70, 43, 15, 11, 12]). To the best of our knowledge, such a penalty approach is not related to Lagrangian relaxation, although this interpretation is natural. Accordingly, the penalty-based algorithms are iterative heuristic methods that do not make use of Lagrangian-based methods. We also do not explore this line of research, but we exploit the main result of Proposition 1 to develop a Benders decomposition method based on the Lagrangian reformulation defined by the objective (4.19) subject to constraints (4.10)-(4.16), i.e., constraints (4.17) and (4.18) are both removed and replaced with the Lagrangian term with sufficiently large values of λ.

4.2. Linear Reformulations

In order to derive linear reformulations, we have to linearize the bilinear terms that appear in (4.17), (4.18) and (4.19). Note that, due to Assumption 3, the only non-linear term contained in the product $u_{la}^s(y,\psi^s)x_{la}^{ks}$ is $\sigma_{ar} y_{ar} x_{la}^{ks}$. These bilinear terms can be linearized by introducing new variables v_{lar}^{ks} and the following constraints:

$$v_{lar}^{ks} \leq x_{la}^{ks}, \quad a \in A, r \in R, s \in S, k \in K, l \in L, \tag{4.22}$$

$$v_{lar}^{ks} \leq \sigma_{ar} y_{ar}, \quad a \in A, r \in R, s \in S, k \in K, l \in L, \tag{4.23}$$

$$v_{lar}^{ks} \geq x_{la}^{ks} - (1 - \sigma_{ar} y_{ar}), \quad a \in A, r \in R, s \in S, k \in K, l \in L, \tag{4.24}$$

$$v_{lar}^{ks} \geq 0, \quad a \in A, r \in R, s \in S, k \in K, l \in L. \tag{4.25}$$

We can then rewrite the objective (4.19) that includes the Lagrangian term ($\mathcal{L}$) as

$$\begin{aligned} Z(\lambda) = &\max \frac{1}{|S|} \sum_{s\in S} \sum_{a\in A} \sum_{r\in R} \sum_{k\in K} \sum_{l\in L} (q_{ar} d_l^k) v_{lar}^{ks} \\ &+ \sum_{s\in S} \sum_{k\in K} \sum_{l\in L} \lambda_l^{ks} \left(\pi_{lD(k)}^{ks} - \sum_{a\in A} \left\{ \sum_{r\in R} \beta_{lar} v_{lar}^{ks} + \psi_{la}^s x_{la}^{ks} \right\} \right). \end{aligned} \tag{4.26}$$

Moreover, we rewrite (4.18), the strong duality constraint ($\mathcal{SD}$), as

$$\pi_{lD(k)}^{ks} = \sum_{a\in A} \left\{ \sum_{r\in R} \beta_{lar} v_{lar}^{ks} + \psi_{la}^s x_{la}^{ks} \right\}, \quad s \in S, k \in K, l \in L. \tag{4.27}$$

The complementary slackness conditions (4.17) are also not linear and contains, in addition to the terms $\sigma_{ar} y_{ar} x_{la}^{ks}$, the products $x_{la}^{ks} \pi_{ln}^{ks}$. Instead of introducing additional variables to represent these products, we exploit the fact that the x variables are, by essence, binary variables and we rewrite the complementary slackness conditions ($\mathcal{CS}$) as

$$u_{la}^s(y,\psi^s) - \pi_{lm}^{ks} + \pi_{ln}^{ks} \leq M(1 - x_{la}^{ks}), \quad a = (n,m) \in A, s \in S, k \in K, l \in L, \tag{4.28}$$

$$x_{la}^{ks} \in \{0,1\}, \quad a \in A, s \in S, k \in K, l \in L, \tag{4.29}$$

where M is an upper bound on the length of any path in the network (due to Assumption 4, we can set $M = |A|\mu$).

We thus obtain a MILP reformulation of our Lagrangian-based single-level non-linear model that has the objective (4.26) subject to constraints (4.10)-(4.16), (2.6)-(4.25) and (4.27)-(4.29). This model combines the three approaches that guarantee optimality of the follower problem: ($\mathcal{CS}$), ($\mathcal{SD}$) and ($\mathcal{L}$), the latter with large enough values of the Lagrange multipliers λ. It is clear that only one of these approaches is sufficient to obtain a reformulation. We therefore consider three MILP models, defined by which of these three elements are included or not. We also consider two additional models that combine one of the two sets of constraints, ($\mathcal{CS}$) or ($\mathcal{SD}$), with the Lagrangian term ($\mathcal{L}$) using large values of λ, in order to measure the impact of the latter on solving the different models. The five resulting MILP models - $\mathscr{M}_{\mathcal{CS}}$, $\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$, $\mathscr{M}_{\mathcal{SD}}$, $\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$ and $\mathscr{M}_{\mathcal{L}}$ - are shown in Table 3.

Model	$\mathcal{CS}$	$\mathcal{SD}$	$\mathcal{L}$
$\mathscr{M}_{\mathcal{CS}}$	✓		
$\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$	✓		✓
$\mathscr{M}_{\mathcal{SD}}$		✓	
$\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$		✓	✓
$\mathscr{M}_{\mathcal{L}}$			✓

Table 3. Definition of the different MILP reformulations

These models can be solved with a state-of-the-art MILP solver. Before presenting computational results that compare the performance of such a MILP solver on the different

models, we develop a Benders decomposition method that exploits the structure of model $\mathscr{M}_{\mathcal{L}}$ to solve large-scale instances of our FCP.

5. Benders Decomposition

In this section, we present a Benders decomposition method designed to solve large-scale instances. The method is based on model $\mathscr{M}_{\mathcal{L}}$ as a starting point because of its particular structure. Since this model does not have neither complementary slackness nor strong duality constraints, it allows for a decomposition that exploits shortest path problem computations separable by scenario and user category.

5.1. Benders Reformulation

Following the partitioning approach of Benders decomposition, we choose the binary variables y to be part of the master problem, while the other, continuous, variables, v, x and π, are handled in the subproblem. Given $y \in Y$, the Benders subproblem decomposes by scenario $s \in S$, by OD pair $k \in K$ and by user category $l \in L$, and further decomposes into a subproblem in variables v and x, and a subproblem in variables π. The latter is exactly the dual of the follower problem (4.4)-(4.6), which is always feasible, as its dual is the bounded shortest path problem (4.1)-(4.3). Hence, only Benders optimality cuts are needed. They are expressed in terms of the extreme points of the polyhedron defined by the flow conservation equations (4.2) and the nonnegativity constraints (4.3), which correspond to the set of paths P^k between $O(k)$ and $D(k)$. If we denote as Π_l^{ks} the variable that approximates the value of the subproblem (4.4)-(4.6), the corresponding Benders optimality cuts are:

$$\Pi_l^{ks} \leq \sum_{a \in A} \left(\sum_{r \in R} \beta_{lar} \sigma_{ar} y_{ar} + \psi_{la}^s \right) \delta_a^p, \quad s \in S, k \in K, l \in L, p \in P^k. \tag{5.1}$$

Concerning the subproblem in variables v and x, we first set

$$\lambda_l^{ks} = \omega \frac{d_l^k}{|S|}, \quad \omega > \frac{\sum_{a \in A} \max_{r \in R} \{\sigma_{ar} q_{ar}\}}{\Delta}$$

in order to derive a convenient expression for the Benders optimality cuts. The subproblem in variables v and x can then be written as:

$$\max \sum_{a \in A} \left\{ \left(\sum_{r \in R} (q_{ar} - \omega \beta_{lar}) v_{lar}^{ks} \right) - \omega \psi_{la}^s x_{la}^{ks} \right\} \tag{5.2}$$

subject to (4.2), (4.3) and

$$v_{lar}^{ks} \leq x_{la}^{ks}, \quad a \in A, r \in R, \quad (\gamma_{lar}^{ks}) \tag{5.3}$$

$$v_{lar}^{ks} \leq \sigma_{ar} y_{ar}, \quad a \in A, r \in R, \quad (\gamma_{lar}^{yks}) \tag{5.4}$$

$$v_{lar}^{ks} \geq x_{la}^{ks} - (1 - \sigma_{ar} y_{ar}), \quad a \in A, r \in R, \quad (\gamma_{lar}^{(1-y)ks}) \tag{5.5}$$

$$v_{lar}^{ks} \geq 0, \quad a \in A, r \in R. \tag{5.6}$$

Note that this subproblem is always feasible, so only Benders optimality cuts are needed.

Using the dual variables shown in parentheses in constraints (5.3)-(5.5), as well as the dual variables θ_{ln}^{ks} associated with the flow conservation equations (4.2), we write the dual of this subproblem to find the expression of the associated Benders optimality cuts:

$$\min \ -\theta_{lD(k)}^{ks} + \sum_{a \in A} \sum_{r \in R} \left(\gamma_{lar}^{yks} \sigma_{ar} y_{ar} + \gamma_{lar}^{(1-y)ks} (1 - \sigma_{ar} y_{ar}) \right) \tag{5.7}$$

$$\theta_{ln}^{ks} - \theta_{lm}^{ks} - \sum_{r \in R} (\gamma_{lar}^{ks} - \gamma_{lar}^{(1-y)ks}) \geq -\omega \psi_{la}^{s}, \quad a = (n,m) \in A, \tag{5.8}$$

$$\gamma_{lar}^{ks} + \gamma_{lar}^{yks} - \gamma_{lar}^{(1-y)ks} \geq q_{ar} - \omega \beta_{lar}, \quad a \in A, r \in R, \tag{5.9}$$

$$\theta_{O(k)ls}^{k} = 0, \tag{5.10}$$

$$\gamma_{lar}^{ks}, \gamma_{lar}^{yks}, \gamma_{lar}^{(1-y)ks} \geq 0, \quad a \in A, r \in R. \tag{5.11}$$

If we denote as w_l^{ks} the variable that approximates the value of the subproblem (5.10)-(5.14), we derive the following Benders optimality cuts:

$$w_l^{ks} \leq -\theta_{lD(k)}^{ks} + \sum_{a \in A} \sum_{r \in R} \left(\gamma_{lar}^{yks} \sigma_{ar} y_{ar} + \gamma_{lar}^{(1-y)ks} (1 - \sigma_{ar} y_{ar}), \right) \tag{5.12}$$

$$s \in S, k \in K, l \in L, (\theta, \gamma, \gamma^y, \gamma^{(1-y)}) \in ext(\mathscr{D}_l^{ks}),$$

where $ext(\mathscr{D}_l^{ks})$ is the set of extreme points of the polyhedron $\mathscr{D}_l^{ks}$ defined by (5.11)-(5.14).

The Benders reformulation $\mathscr{M}_{\mathcal{BD}}$ can then be written as:

$$\max \frac{1}{|S|} \sum_{s \in S} \sum_{k \in K} \sum_{l \in L} d_l^k \left(w_l^{ks} + \omega \Pi_l^{ks} \right) \tag{5.13}$$

subject to (4.10)-(4.12), (5.1), (5.12) and

$$Z_0^l \leq \frac{1}{|S|} \sum_{s \in S} \sum_{k \in K} \sum_{l \in L} d_l^k \left(w_l^{ks} + \omega \Pi_l^{ks} \right) \leq Z_0^u, \tag{5.14}$$

where Z_0^l and Z_0^u are, respectively, lower and upper bounds on the optimal objective value that are obtained by performing the initial heuristic method described in Section 5.3 and by solving the initial relaxation presented in Section 5.4. Before we look into the computation of these initial bounds, we first study how to generate efficiently the Benders optimality cuts (5.1) and (5.12).

5.2. Generation of Benders Cuts

As Benders decomposition is applied first to the linear programming (LP) relaxation (following the pioneering work of [53]), we show how to generate Benders cuts for $\overline{y} \in \overline{Y}$, where

$$\overline{Y} = \left\{ y \in [0,1]^{|A| \times |R|} \mid \sum_{a \in A} \sum_{r \in R} \sigma_{ar} c_{ar} y_{ar} \leq b; \sum_{r \in R} \sigma_{ar} y_{ar} \leq 1, a \in A \right\},$$

specializing them subsequently for $\overline{y} \in Y$. In particular, we show that the Benders cuts can be generated efficiently by solving shortest path problems, and by adding a "small" additional effort when $\overline{y}$ is fractional, and no additional effort at all when $\overline{y}$ is integral. Given $\overline{y} \in \overline{Y}$, the Benders subproblem decomposes by $s \in S$, $k \in K$ and $l \in L$. We first look at the solution of the dual of the subproblem in variables v and x, i.e., (5.10)-(5.14). To simplify the notation in what follows, we define $\xi_{lar} \equiv q_{ar} - \omega\beta_{lar}$, $l \in L$, $a \in A$, $r \in R$.

Proposition 2. *For $s \in S$, $k \in K$ and $l \in L$, let $(\overline{\theta}, \overline{\gamma}, \overline{\gamma}^y, \overline{\gamma}^{(1-y)})$ be defined as*

$$\overline{\gamma}^{ks}_{lar} = \max\{0, \xi_{lar}\}\sigma_{ar}\overline{y}_{ar}, \quad a \in A, r \in R, \tag{5.15}$$

$$\overline{\gamma}^{yks}_{lar} = \max\{0, \xi_{lar}\}(1 - \sigma_{ar}\overline{y}_{ar}), \quad a \in A, r \in R, \tag{5.16}$$

$$\overline{\gamma}^{(1-y)ks}_{lar} = \max\{0, -\xi_{lar}\}\sigma_{ar}\overline{y}_{ar}, \quad a \in A, r \in R, \tag{5.17}$$

$$-\overline{\theta}^{ks}_{l} = (-\overline{\theta}^{ks}_{ln})_{n\in N} \text{ solves the linear program:} \tag{5.18}$$

$$\max \pi^{ks}_{lD(k)}$$

$$\pi^{ks}_{lm} - \pi^{ks}_{ln} \leq \left(\omega u^s_{la}(\overline{y},\psi^s) - \sum_{r\in R} q_{ar}\sigma_{ar}\overline{y}_{ar}\right), \quad a = (n,m) \in A,$$

$$\pi^{ks}_{lO(k)} = 0.$$

(1) For any $\overline{y} \in \overline{Y}$, $(\overline{\theta}, \overline{\gamma}, \overline{\gamma}^y, \overline{\gamma}^{(1-y)})$ is feasible for the subproblem (5.10)-(5.14).

(2) For any $\overline{y} \in Y$, $(\overline{\theta}, \overline{\gamma}, \overline{\gamma}^y, \overline{\gamma}^{(1-y)})$ is optimal for the subproblem (5.10)-(5.14) when $y = \overline{y}$.

PROOF. For any $\overline{y} \in \overline{Y}$, we note the following identity, which follows directly from (5.15) and (5.17):

$$\overline{\gamma}^{ks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar} = \xi_{lar}\sigma_{ar}\overline{y}_{ar}. \tag{5.19}$$

We first prove that the solution given by (5.15)-(5.18) is feasible for the subproblem (5.10)-(5.14). It follows from (5.18) that $\overline{\theta}^{ks}_l$ verifies, for $a = (n,m) \in A$:

$$\begin{aligned}
\overline{\theta}^{ks}_{ln} - \overline{\theta}^{ks}_{lm} &\geq \sum_{r\in R}(q_{ar} - \omega\beta_{lar})\sigma_{ar}\overline{y}_{ar} - \omega\psi^s_{la}, \\
&= \sum_{r\in R}\xi_{lar}\sigma_{ar}\overline{y}_{ar} - \omega\psi^s_{la}, \\
&= \sum_{r\in R}(\overline{\gamma}^{ks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar}) - \omega\psi^s_{la},
\end{aligned}$$

which immediately implies (5.11). To verify (5.12), let $a \in A$ and $r \in R$, and consider the two cases:

(1) $\xi_{lar} > 0$, then

$$\begin{aligned}\overline{\gamma}^{ks}_{lar} + \overline{\gamma}^{yks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar} &= (\overline{\gamma}^{ks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar}) + \overline{\gamma}^{yks}_{lar} \\ &= \xi_{lar}\sigma_{ar}\overline{y}_{ar} + \xi_{lar}(1 - \sigma_{ar}\overline{y}_{ar}) \\ &= \xi_{lar} \\ &= q_{ar} - \omega\beta_{lar}.\end{aligned}$$

(2) $\xi_{lar} \leq 0$, then

$$\begin{aligned}\overline{\gamma}^{ks}_{lar} + \overline{\gamma}^{yks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar} &= (\overline{\gamma}^{ks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar}) + \overline{\gamma}^{yks}_{lar} \\ &= \xi_{lar}\sigma_{ar}\overline{y}_{ar} + 0 \\ &= (q_{ar} - \omega\beta_{lar})\sigma_{ar}\overline{y}_{ar} \\ &\geq q_{ar} - \omega\beta_{lar}.\end{aligned}$$

Finally, constraints (5.13) are (5.14) are trivially verified.

Now, consider $\overline{y} \in Y$ and let us show that the solution given by (5.15)-(5.19) is optimal for the subproblem (5.10)-(5.14) when $y = \overline{y}$. The complementary slackness conditions can be written as:

$$\gamma^{ks}_{lar}\left(x^{ks}_{la} - v^{ks}_{lar}\right) = 0, \quad a \in A, r \in R, \tag{5.20}$$

$$\gamma^{yks}_{lar}\left(\sigma_{ar}\overline{y}_{ar} - v^{ks}_{lar}\right) = 0, \quad a \in A, r \in R, \tag{5.21}$$

$$\gamma^{(1-y)k}_{lar}\left(v^{ks}_{lar} - x^{ks}_{la} + (1 - \sigma_{ar}\overline{y}_{ar})\right) = 0, \quad a \in A, r \in R, \tag{5.22}$$

$$x^{ks}_{la}\left(\theta^{ks}_{ln} - \theta^{ks}_{lm} - \sum_{r\in R}(\gamma^{ks}_{lar} - \gamma^{(1-y)ks}_{lar}) + \omega\psi^{s}_{la}\right) = 0, \quad a = (n,m) \in A, \tag{5.23}$$

$$v^{ks}_{lar}\left(\gamma^{ks}_{lar} + \gamma^{yks}_{lar} - \gamma^{(1-y)ks}_{lar} - (q_{ar} - \omega\beta_{lar})\right) = 0, \quad a \in A, r \in R. \tag{5.24}$$

Let $\overline{x}^{ks}_{l} = (\overline{x}^{ks}_{la})_{a\in A}$ denote the solution of the shortest path problem between $O(k)$ and $D(k)$ with respect to arc lengths $(\omega u^{s}_{la}(\overline{y},\psi^{s}) - \sum_{r\in R} q_{ar}\sigma_{ar}\overline{y}_{ar})$. By (5.18), $-\overline{\theta}^{ks}_{l}$ solves the dual of this shortest path problem. Hence, the following holds for $a = (n,m) \in A$:

$$\begin{aligned}0 &= \overline{x}^{ks}_{la}\left(\overline{\theta}^{ks}_{ln} - \overline{\theta}^{ks}_{lm} + \left(\omega u^{s}_{la}(\overline{y},\psi^{s}) - \sum_{r\in R} q_{ar}\sigma_{ar}\overline{y}_{ar}\right)\right) \\ &= \overline{x}^{ks}_{la}\left(\overline{\theta}^{ks}_{ln} - \overline{\theta}^{ks}_{lm} - \sum_{r\in R}(q_{ar} - \omega\beta_{lar})\sigma_{ar}\overline{y}_{ar} + \omega\psi^{s}_{la}\right) \\ &= \overline{x}^{ks}_{la}\left(\overline{\theta}^{ks}_{ln} - \overline{\theta}^{ks}_{lm} - \sum_{r\in R}\xi_{lar}\sigma_{ar}\overline{y}_{ar} + \omega\psi^{s}_{la}\right) \\ &= \overline{x}^{ks}_{la}\left(\overline{\theta}^{ks}_{ln} - \overline{\theta}^{ks}_{lm} - \sum_{r\in R}(\overline{\gamma}^{ks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar}) + \omega\psi^{s}_{la}\right),\end{aligned}$$

and (5.23) is verified. To show that the other complementary slackness conditions are satisfied, we consider two cases, for any $a \in A$ and $r \in R$:

(1) $\sigma_{ar}\overline{y}_{ar} = 1$, which implies that $\overline{\gamma}^{yks}_{lar} = 0$, so (5.21) is verified. In addition, we must have $\overline{v}^{ks}_{lar} = \overline{x}^{ks}_{la}$ to satisfy (5.3)-(5.6), so (5.20) and (5.22) are verified. Finally, $\overline{\gamma}^{ks}_{lar} + \overline{\gamma}^{yks}_{lar} - \overline{\gamma}^{(1-y)ks}_{lar} = q_{ar} - \omega\beta_{lar}$ and (5.24) is verified.

(2) $\sigma_{ar}\overline{y}_{ar} = 0$, which implies that $\overline{\gamma}^{ks}_{lar} = \overline{\gamma}^{(1-y)ks}_{lar} = 0$, so (5.20) and (5.22) are verified. In addition, we must have $\overline{v}^{ks}_{lar} = 0$ to satisfy (5.3)-(5.6), so (5.21) and (5.24) are verified.

□

It is interesting to note that, in the case where $\overline{y} \in Y$, i.e., $\overline{y}$ is integral, and we then use the optimal dual solution $(\overline{\theta}, \overline{\gamma}, \overline{\gamma}^y, \overline{\gamma}^{(1-y)})$, the Benders optimality cuts (5.12) can be simplified using the following identities:

$$\begin{aligned}\sum_{a\in A}\sum_{r\in R}\overline{\gamma}^{yks}_{lar}\sigma_{ar}y_{ar} &= \sum_{a\in A}\sum_{r\in R}\max\{0,\xi_{lar}\}(1-\sigma_{ar}\overline{y}_{ar})\sigma_{ar}y_{ar}\\ &= \sum_{a\in A, r\in R|\xi_{lar}>0}\xi_{lar}(1-\sigma_{ar}\overline{y}_{ar})\sigma_{ar}y_{ar}\end{aligned}$$

and

$$\begin{aligned}\sum_{a\in A}\sum_{r\in R}\overline{\gamma}^{(1-y)ks}_{lar}(1-\sigma_{ar}y_{ar}) &= \sum_{a\in A}\sum_{r\in R}\max\{0,-\xi_{lar}\}\sigma_{ar}\overline{y}_{ar}(1-\sigma_{ar}y_{ar})\\ &= \sum_{a\in A, r\in R|\xi_{lar}<0} -\xi_{lar}\sigma_{ar}\overline{y}_{ar}(1-\sigma_{ar}y_{ar}).\end{aligned}$$

Thus, when $\overline{y}$ is integral, any of the Benders optimality cuts (5.12) can be written as:

$$\begin{aligned}w^{ks}_l \le -\overline{\theta}^{ks}_{lD(k)} &+ \sum_{a\in A, r\in R|\xi_{lar}>0}\xi_{lar}(1-\sigma_{ar}\overline{y}_{ar})\sigma_{ar}y_{ar}\\ &+ \sum_{a\in A, r\in R|\xi_{lar}<0} -\xi_{lar}\sigma_{ar}\overline{y}_{ar}(1-\sigma_{ar}y_{ar}).\end{aligned}$$

We can exploit Assumption 3 to further simplify the expression of the Benders optimality cuts (5.12).

Let us assume that category $l \in L$ contains only cooperative users. We then have $\beta_{lar} \le 0$, which implies $\xi_{lar} > 0$, for any $a \in A$ and $r \in R$. The corresponding cut reduces to

$$w^{ks}_l \le -\overline{\theta}^{ks}_{lD(k)} + \sum_{a\in A, r\in R|\xi_{lar}>0}\xi_{lar}(1-\sigma_{ar}\overline{y}_{ar})\sigma_{ar}y_{ar},$$

or, equivalently, to

$$w^{ks}_l \le -\overline{\theta}^{ks}_{lD(k)} + \sum_{a\in A, r\in R|\xi_{lar}>0, \sigma_{ar}\overline{y}_{ar}=0}\xi_{lar}\sigma_{ar}y_{ar}.$$

This cut has a simple interpretation: if the current approximation $\overline{w}_l^{ks}$ of the value of the subproblem (5.10)-(5.14) does not correspond with the actual shortest path length $-\overline{\theta}_{lD(k)}^{ks}$ when the installed resources are given by $\overline{y}$, i.e., $\overline{w}_l^{ks} > -\overline{\theta}_{lD(k)}^{ks}$, then the cut is activated. Moreover, in this case, the cut provides an incentive to install additional resources that could decrease the disutility of some of the users (and, at the same time, increase the captured flow), but are currently not used, i.e., resources $r \in R$ on arcs $a \in A$ such that $\sigma_{ar}\overline{y}_{ar} = 0$.

In a similar way, if category $l \in L$ contains only strongly evasive users, we then have $\beta_{lar} > 0$ for any arc $a \in A$ and resource type $r \in R$. Given that ω is typically large, we may assume that $\beta_{lar} > 1/\omega$, which implies that $\xi_{lar} < 0$, for any $a \in A$ and $r \in R$. The Benders optimality cut then simplifies to

$$w_l^{ks} \leq -\overline{\theta}_{lD(k)}^{ks} + \sum_{a \in A, r \in R | \xi_{lar} < 0} -\xi_{lar}\sigma_{ar}\overline{y}_{ar}(1 - \sigma_{ar}y_{ar}),$$

or, equivalently, to

$$w_l^{ks} \leq -\overline{\theta}_{lD(k)}^{ks} + \sum_{a \in A, r \in R | \xi_{lar} < 0, \sigma_{ar}\overline{y}_{ar} = 1} -\xi_{lar}(1 - \sigma_{ar}y_{ar}).$$

In this case, the cut provides an incentive to close resources $r \in R$ that are currently installed on some arcs $a \in A$, i.e., $\sigma_{ar}\overline{y}_{ar} = 1$, but could be closed to decrease the disutility of some of the users.

Proposition 2 is used to generate Benders cuts. Given $\overline{y} \in \overline{Y}$, shortest path problems with respect to arc lengths $(\omega u_{la}^s(\overline{y},\psi^s) - \sum_{r \in R} q_{ar}\sigma_{ar}\overline{y}_{ar})$ are solved for each scenario $s \in S$ and each user category $l \in L$. At most $|N|$ applications of Dijkstra's algorithm are needed to identify the shortest paths for each OD pair $k \in K$. Note that these paths are also the shortest ones with respect to arc lengths $u_{la}^s(\overline{y},\psi^s)$, so we can use these paths to generate the Benders optimality cuts (5.1) associated with the subproblem in variables π. The other Benders optimality cuts, (5.12), are generated by using the result of Proposition 2. If $\overline{y} \in Y$, the cuts are generated based on the solution $(\overline{\theta}, \overline{\gamma}, \overline{\gamma}^y, \overline{\gamma}^{(1-y)})$. If $\overline{y} \in \overline{Y}$, the solution $(\overline{\theta}, \overline{\gamma}, \overline{\gamma}^y, \overline{\gamma}^{(1-y)})$ is given as input to the linear program (5.10)-(5.14), which is then solved to derive the corresponding Benders cuts (5.12).

5.3. Initial Heuristic

We use a heuristic to provide initial cuts to our Benders reformulation, as well as to find an initial solution of value Z_0^l . At every iteration of the heuristic, we solve the 0-1 LP model

$$\max_{y \in Y} \sum_{a \in A} \sum_{r \in R} f_a q_{ar} \sigma_{ar} y_{ar} \tag{5.25}$$

where f_a is an estimate of the traffic flow using arc a. This estimate combines the traffic flows for every user category and every OD pair. Given an optimal solution $\overline{y} \in Y$, the follower problem (3.13) for $y = \overline{y}$ is solved with Dijkstra's algorithm, thus identifying a feasible

solution ($\overline{x}$,$\overline{y}$) to our FCP and a lower bound Z^l. At the next iteration, f_a is updated as follows:

$$f_a = \frac{1}{|S|} \sum_{s \in S} \sum_{k \in K} \sum_{l \in L} d_l^k \overline{x}_{al}^{ks}, \quad a \in A. \tag{5.26}$$

This procedure is repeated until the same lower bound is found for two consecutive iterations or until a maximum number of iterations is reached.

We perform this iterative procedure five times by changing the initial value of f_a, which is set to $f_a = \eta \sum_{k \in K} \sum_{l \in L} d_l^k$, where $\eta \in \{0.2, 0.4, 0.6, 0.8, 1\}$. The best feasible solution found, along with its value Z_0^l, is then given as input to the Benders decomposition method. To generate initial Benders cuts, we use as candidate solutions, each optimal solution $\overline{y} \in Y$ obtained when solving (5.25) at every iteration of the heuristic.

5.4. Initial Relaxation

We define an initial relaxation that is valid only when the strong homogeneity assumption (see Section 3) holds, i.e., all users in any category are either strongly cooperative or strongly evasive (and there are no indifferent users). If this assumption does not hold, then we could solve the LP relaxation of any of the five MILP models presented in Section 4.2 to compute the upper bound Z_0^u provided to the Benders reformulation.

The relaxation we propose exploits the Lagrangian-based single-level non-linear model. Our goal is to get rid of the products $\sigma_{ar} y_{ar} x_{la}^{ks}$ and, consequently, of the many variables v_{lar}^{ks} that are generated when linearizing these products. The resulting relaxation would then be easier to solve than any LP relaxation derived from the five MILP models presented in Section 4.2. Under the strong homogeneity assumption, we use this relaxation to compute an initial upper bound Z_0^u to give to the Benders reformulation.

For each user category $l \in L$, we define τ_l: $= -1$, if users in category l are strongly cooperative; $= +1$, is users in category l are strongly evasive.

Proposition 3. *For each $l \in L$, let*

$$\zeta_l = \tau_l \max_{a \in A, r \in R | \sigma_{ar} = 1} \left\{ \frac{q_{ar}}{\beta_{lar}} \right\}.$$

The following MILP model provides an upper bound on Z:

$$Z_0^u = \max \frac{1}{|S|} \sum_{s \in S} \sum_{k \in K} \sum_{l \in L} \zeta_l d_l^k \left(\pi_{lD(k)}^{ks} - \sum_{a \in A} \psi_{la}^s x_{la}^{ks} \right) \tag{5.27}$$

subject to (4.10)-(4.16) and (4.28)-(4.29).

Proof. By definition of ζ_l, we have, for each $l \in L$, $q_{ar} - \zeta_l \beta_{lar} \leq 0$, for each $a \in A$ and $r \in R$ such that $\sigma_{ar} = 1$. If we set the Lagrange multipliers to the values $\lambda_l^{ks} = \zeta_l \frac{d_l^k}{|S|}$, for $s \in S$, $k \in K$ and $l \in L$, we can write the objective of the Lagrangian-based single-level

non-linear model as

$$\max \frac{1}{|S|} \sum_{s\in S} \sum_{k\in K} \sum_{l\in L} d_l^k \left(\sum_{a\in A} \sum_{r\in R} \underbrace{\left(q_{ar} - \zeta_l \beta_{lar}\right) \sigma_{ar} y_{ar} x_{la}^{ks}}_{\leq 0} \right)$$
$$+ \frac{1}{|S|} \sum_{s\in S} \sum_{k\in K} \sum_{l\in L} \zeta_l d_l^k \left(\pi_{lD(k)}^{ks} - \sum_{a\in A} \psi_{la}^s x_{la}^{ks} \right),$$

which implies the result. □

Since this relaxation is a MILP model, we solve it at the root node of a branch-and-cut algorithm implemented in a state-of-the-art solver.

5.5. Summary of the Algorithm

The Benders reformulation is solved by Branch-and-Benders-Cut (BBC), where a single branch-and-cut tree is generated with cuts being added as it is explored. The classical approach to solve a Benders decomposition that consists in solving a MILP master problem at every iteration and adding cuts in between is nowadays considered to be inferior to such a BBC approach.

Our BBC algorithm proceeds in two phases:

Initialization phase

(1) Solve the initial relaxation presented in Section 5.4 to obtain Z_0^u.

(2) Perform the initial heuristic presented in Section 5.3 to:
   (a) Find an initial feasible solution of value Z_0^l;
   (b) Generate the cuts on w_l^{ks} and Π_l^{ks} from every solution $\overline{y} \in Y$ obtained when solving (5.25) and add them to the Benders reformulation.

(3) Add constraint (5.14) to the Benders reformulation.

Branch-and-cut phase

(4) At the root node:
   (a) Solve the LP relaxation of the master problem to obtain $\overline{y} \in \overline{Y}$;
   (b) Solve shortest path problems with respect to arc legths $(\omega u_{la}^s(\overline{y},\psi^s) - \sum_{r\in R} q_{ar}\sigma_{ar}\overline{y}_{ar})$;
   (c) Generate Benders cuts (5.1) on the values of Π_l^{ks};
   (d) If $\overline{y}$ is integral, generate Benders cuts (5.12) on the values of w_l^{ks};
   (e) If $\overline{y}$ is fractional, solve the subproblem (5.10)-(5.14) to generate Benders cuts on the values of w_l^{ks}.

(5) At a node corresponding to an integer solution $\overline{y} \in Y$:
   (a) Solve shortest path problems with respect to arc legths $(\omega u_{la}^s(\overline{y},\psi^s) - \sum_{r\in R} q_{ar}\sigma_{ar}\overline{y}_{ar})$;
   (b) Generate Benders cuts (5.1) on the values of Π_l^{ks};

(c) Generate Benders cuts (5.12) on the values of w_l^{ks}.

6. Computational Experiments

There are two main objectives guiding our computational experiments. The first is to assess the impact that various problem characteristics have when solving each of the models presented with a state-of-the-art MILP solver. These characteristics include evasive and cooperative user behaviors and levels of budget. The second objective is to test the scalability of each model when problem dimensions are increased. The dimensions we focus on are the number of scenarios, the number of OD pairs and the number of candidate arcs.

The five models presented in Section 4.2 are solved with the branch-and-cut algorithm of CPLEX (version 12.7.1.0), using default parameters. In addition, the best feasible solution found by the heuristic presented in Section 5.3 is initially provided when solving each model. The Benders decomposition method is also implemented with CPLEX. We use the user cut callback to implement the generation of Benders cuts at the root node and the lazy cut callback to implement the generation of Benders cuts at nodes where an integer solution is found.

Our experiments are conducted on two distinct networks. One is a small network specifically designed to study the impact of the problem characteristics and the other is a large network where the computational limits of the models can be tested. For both sets of experiments, the disutility of any arc $a \in A$ for a user category $l \in L$ is based on the presence of resources on the arc, the length of the arc t_a and the random term ε_{la}. It is defined as

$$u_{la}(y, t, \varepsilon) = \sum_{r \in R} \beta_{lar} \sigma_{ar} y_{ar} + \alpha_l t_a + \varepsilon_{la}.$$

Furthermore, the discrete choice model used is a nested recursive logit model as it is a superior alternative to the popular logit model [47]. This was done by defining the distribution scale of ε_{la} in (3.10) as proportional to t_a and by adding a constant κ, which penalizes paths that have a high number of arcs, as κ is added to the disutility of each arc. We thus use the following definition for the random term:

$$\varepsilon_{al} = \text{GEV}(0, e^{\theta t_a}) + \kappa,$$

where GEV stands for generalized extreme value distribution with two parameters. The first parameter is the location, 0 in our case, and the second is the scale, which is given by the exponential function with exponent θ, a non-negative value, multiplied by the length of arc a. Values are sampled from the distribution object extreme_value_distribution in the C++ library. We note that various discrete choice models could have been used by simply changing the definition of ε_{la}.

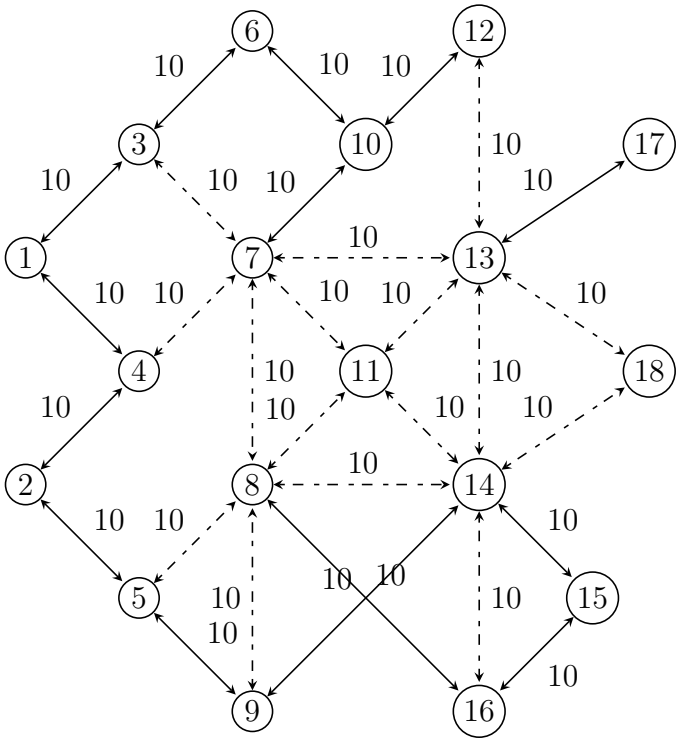


Fig. 5. A small network

All experiments were conducted on a machine with an Intel(R) Xeon(R) CPU E3-1275 V2 @ 3.50GHz along with 28GB of RAM.

6.1. Small Network

We begin our numerical tests on a small network comprised of 18 nodes and 60 arcs illustrated in Figure 5. Instances based on this network have 8 OD pairs and 32 candidate arcs. These arcs are indicated by dashed lines. There are 2 resource types and thus 64 possible resources, 1 user category and 30 scenarios. The number of scenarios has been determined following preliminary experiments. The value $|S| = 30$ ensures a stable solution, i.e., the solution does not change when increasing further the number of scenarios. The 8 OD pairs, (1-18), (2-17), (6-16), (9-12), (18-1), (17-2), (16-6) and (12-9), have an equal aggregate demand of 10 and promote traffic flow interactions in the central area of the network where resources can be installed. This is a key element in ensuring that the instances are not trivial.

Four different types of user behaviors are tested on the small network: mildly evasive (ME), very evasive (VE), mildly cooperative (MC) and very cooperative (VC). Table 4 summarizes the parameters related to the two types of resources ($r = 1,2$) and to the part of the disutility of an arc that depends on user category preferences. Column "β_{lar}" contains the values for each type of user behavior with respect to each of the two types of resources.

r	β_{lar}				q_{ar}	c_{ar}
	ME	VE	MC	VC		
1	30	60	-30	-60	0.6	1
2	20	40	-20	-40	0.4	1

Table 4. Parameters for the two resource types for the small network ($\alpha_l = 10$)

Columns "q_{ar}" and "c_{ar}" represent, respectively, the corresponding values for each type of resources. Parameter α_l has the same value of 10 for every type of user behavior.

The parameters related to the random term ε_{la} take on the following values: $\theta = 0.08$ and $\kappa = 5$. This calibration of the parameters is designed so that the shortest path between each OD pair varies based on the random term ε_{la} and the presence of control resources. This further guarantees non-trivial instances.

The goal of these tests is to identify instance characteristics that make the problem harder to solve as well as to evaluate the differences between the various models. For each of the four types of user behavior, four levels of budget are tested, allowing 3 (10%), 8 (25%), 16 (50%) or 24 (75%) out of the 32 candidate arcs where resources can be installed.

Table 5 reports three values for each model tested on each instance. First, "time" indicates the time in seconds either to find an optimal solution or to stop after a time limit of 1 hour. Second, "gap_r" indicates the gap in percentage at the root node between the best relaxation bound and the optimal value. Third, "gap_f" indicates the gap in percentage between the value of the final integer solution and the optimal value. Also, "gap_h", the gap in percentage between the solution found by the heuristic presented in Section 5.3 and the optimal value, is indicated for each instance. If none of the models could find the optimal solution in the allotted time, then the best solution found across all models is used in place of an optimal solution.

Our heuristic provides initial solutions of varying quality. As can be expected, when users try to avoid resources, the iterative nature of the heuristic lacks the foresight needed to capture traffic flows. When resources are installed on the arcs with the most flow, users change their path choice to avoid them, which causes the heuristic to reassign the resources to the new shortest paths, perpetuating the cycle. This effect becomes most apparent in the very evasive case where solutions are relatively poor. In the cooperative case, however, the leader and the follower problems are "aligned", in that traffic flows are attracted to the installed resources. This is reflected in the mildly cooperative case where the optimal solution is found for every budget value. Nonetheless, the heuristic can struggle to find good solutions with cooperative users as can be seen in the very cooperative case. This is due to the greedy nature of the approach in which the arcs with the most flow initially heavily dictate any solution found as the traffic reassignments only compound what is already captured.

		$\mathscr{M}_{\mathcal{CS}}$	$\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$	$\mathscr{M}_{\mathcal{SD}}$	$\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$	$\mathscr{M}_{\mathcal{L}}$	$\mathscr{M}_{\mathcal{BD}}$
ME-10%	time [s]	4.92	9.72	29.4	9.74	351.7	124.65
	gap_r [%]	12.98	53.33	0	0	53.33	53.33
gap_h=20.00%	gap_f [%]	0	0	0	0	0	0
ME-25%	time [s]	35.85	19.81	37.7	12.2	842.62	518.4
	gap_r [%]	18.86	51.67	0.19	0	51.67	51.67
gap_h=10.00%	gap_f [%]	0	0	0	0	0	0
ME-50%	time [s]	44.48	20.54	36.11	10.83	1002.83	537.79
	gap_r [%]	13.34	51.04	0	0	51.04	51.04
gap_h=0%	gap_f [%]	0	0	0	0	0	0
ME-75%	time [s]	12.51	19.62	16.12	10.85	902.15	730.02
	gap_r [%]	18.13	51.04	0	0	51.04	51.04
gap_h=0%	gap_f [%]	0	0	0	0	0	0
VE-10%	time [s]	42.46	1759.72	19.86	23.25	952.44	807.57
	gap_r [%]	14.91	53.33	0	0	53.33	53.33
gap_h=40.00%	gap_f [%]	0	0	0	0	0	0
VE-25%	time [s]	934.29	3600	31.65	18.56	3600	3600
	gap_r [%]	31.85	51.67	0	0	51.67	51.67
gap_h=20.00%	gap_f [%]	0	0	0	0	10.00	15.00
VE-50%	time [s]	1963.64	3600	52.59	49.2	3600	3600
	gap_r [%]	42.94	51.11	2.79	2.82	51.11	51.11
gap_h=60.00%	gap_f [%]	0	0	0	0	2.22	6.67
VE-75%	time [s]	3281.43	3600	19.87	24.04	3600	3600
	gap_r [%]	51.04	51.04	0	0	51.04	51.04
gap_h=12.50%	gap_f [%]	0	0	0	0	0	0
MC-10%	time [s]	3.38	83.19	692.07	739.63	3600	3600
	gap_r [%]	0	52.78	19.26	19.26	52.78	52.78
gap_h=0%	gap_f [%]	0	0	0	0	0	0
MC-25%	time [s]	48.5	216	3600	3600	3600	3600
	gap_r [%]	18.96	51.39	51.39	51.39	51.39	51.39
gap_h=0%	gap_f [%]	0	0	0	0	0	0
MC-50%	time [s]	98.67	183.84	3600	3600	3600	3600
	gap_r [%]	19.85	50.93	50.93	50.93	50.93	50.93
gap_h=0%	gap_f [%]	0	0	0	0	0	0
MC-75%	time [s]	42.49	174.16	3600	3600	3600	3600
	gap_r [%]	29.36	50.93	50.93	50.93	50.93	50.93
gap_h=0%	gap_f [%]	0	0	0	0	0	0
VC-10%	time [s]	3600	3600	3600	3600	3600	3600
	gap_r [%]	127.02	1196.22	112.09	112.09	1196.22	1196.22
gap_h=24.37%	gap_f [%]	0	24.37	13.45	12.61	24.37	24.37
VC-25%	time [s]	3600	3600	3600	3600	3600	3600
	gap_r [%]	105.58	502.99	136.91	136.91	502.99	502.99
gap_h=10.56%	gap_f [%]	10.56	10.21	0.70	0	10.56	10.56
VC-50%	time [s]	3600	3600	3600	3600	3600	3600
	gap_r [%]	121.76	353.87	203.19	203.18	353.87	353.87
gap_h=17.71%	gap_f [%]	17.71	17.71	17.71	0	17.71	17.71
VC-75%	time [s]	3600	3600	3600	3600	3600	3600
	gap_r [%]	110.84	276.28	201.40	202.00	276.28	276.28
gap_h=14.11%	gap_f [%]	14.11	14.11	7.98	0	14.11	14.11

Table 5. Small network results with 30 scenarios

As such, better solutions with resources installed on arcs with little to no flow initially are unlikely to be found.

Our results show that, in general, it is more difficult to prove optimality for the instances with cooperative users than for the ones with evasive users. The values of "gap_r" indicate that the relaxations are tighter for the last type of instances. As a result, installing a resource that repels traffic flow allows for earlier pruning in the branch-and-cut tree. Indeed, if a path is not used when some resources are installed on it, then there is no need to explore installing additional resources on it. However, when resources make a path more interesting for users, the same logic cannot be applied, since adding resources could attract more users and therefore yield better solutions.

We can also see that a budget allowing for a little over 25% of the possible resources to be installed seems to entail the most difficulty. If the budget is too small, there are few possibilities to consider, while if the budget is too large, there are no tradeoffs to be made.

Turning our attention to our various formulations, we notice some distinctive results. For evasive behavior cases, the aggregated models $\mathscr{M}_{\mathcal{SD}}$ and $\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$ seem to be the best, while other models struggle. In the mildly cooperative case, however, models $\mathscr{M}_{\mathcal{CS}}$ and $\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$ show better results than the others. The very cooperative case seems to be too difficult for any of the models. Overall, we notice that $\mathscr{M}_{\mathcal{BD}}$ has better results than $\mathscr{M}_{\mathcal{L}}$, which confirms that applying our Benders decomposition to $\mathscr{M}_{\mathcal{L}}$ is indeed an improvement to that formulation. However, due to the relatively small size of these instances, $\mathscr{M}_{\mathcal{L}}$ does not perform nearly as well as $\mathscr{M}_{\mathcal{SD}}$ and $\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$ for the evasive behavior cases, and $\mathscr{M}_{\mathcal{CS}}$ and $\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$ for the cooperative behavior cases. Adding the Lagrangian term to models $\mathscr{M}_{\mathcal{CS}}$ and $\mathscr{M}_{\mathcal{CSD}}$ seems to help in the evasive cases with budget values of 25% and 50%. With these budget values identified as more difficult, it is reasonable to conclude that the Lagrangian term only helps for instances that have a minimum of difficulty. If the instance is easy to solve, the additional computational burden caused by the Lagrangian term can lead to longer solution times.

6.2. Winnipeg network

The second network is that of the city of Winnipeg and is comprised of 1040 nodes and 2836 arcs. As with our experiments on the small network, these instances are designed to promote interactions between flows and control resources as much as possible. There are 2 resource types and 1 user category. Candidate arcs are selected by decreasing order of flows from the best known flow solution included in the network description files available at https://github.com/bstabler/TransportationNetworks/tree/master/Winnipeg

OD pairs are randomly sampled from the complete list provided with the network files as to spread out traffic flows within the network. The budget is fixed at 30% meaning that 30% of all possible resources can be installed at most. The number of scenarios, OD pairs and

r	β_{lar}	q_{ar}	c_{ar}
1	0.6	0.6	1
2	0.4	0.4	1

Table 6. Parameters for the two types of resources for the Winnipeg network ($\alpha_l = 0.001$)

candidate arcs varies throughout the experiments and is therefore specified wherever results are reported.

As the focus of the experiments on this much larger network shifts towards scalability, we only test an evasive user behavior, for which it is more difficult to find good feasible solutions. This also corresponds to what is mostly studied in the literature. Table 6 summarizes the parameters related to the two types of resources (r = 1,2) and to the part of the disutility of an arc that depends on user category preferences. The columns are defined in the same way as for Table 4.

The parameters related to the random term ε_{la} take on the following values: $\theta = 0.0001$ and $\kappa = 4.5$. Similarly to the small network, this calibration of the parameters helps promote changing shortest paths depending on the installation of resources and on the random term.

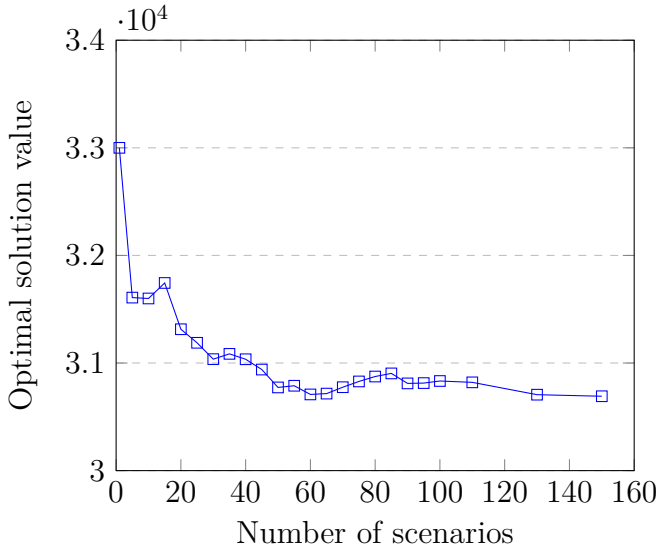
$\|S\|$	$\mathscr{M}_{\mathcal{CS}}$	$\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$	$\mathscr{M}_{\mathcal{SD}}$	$\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$	$\mathscr{M}_{\mathcal{L}}$	$\mathscr{M}_{\mathcal{BD}}$
1	7200	7200	13	12	17	66
15	7200	7200	2818	3512	7200	627
30	7200	7200	7200	7200	7200	1573

Table 7. Average solution times (5 instances), 50 candidate arcs, 40 OD pairs

The first set of experiments aims to show how well each model performs when the network is very large. Table 7 shows average solution times over 5 instances with 50 candidate arcs, 40 OD pairs and $|S|$ = 1,15, 30. All six models are solved with a time limit set at 2 hours. We can see that $\mathscr{M}_{\mathcal{CS}}$ and $\mathscr{M}_{\mathcal{CS}-\mathcal{L}}$ are unable to solve any of the instances, as the complementary slackness conditions make the models too large even for 1 scenario. Models $\mathscr{M}_{\mathcal{SD}}$, $\mathscr{M}_{\mathcal{SD}-\mathcal{L}}$ and $\mathscr{M}_{\mathcal{L}}$ are the fastest for 1 scenario, but model $\mathscr{M}_{\mathcal{BD}}$ overtakes them when $|S| = 15$ and even more when $|S| = 30$. Subsequent experiments, which explore solution stability and the effects of increasing problem dimensions, are only conducted with model $\mathscr{M}_{\mathcal{BD}}$ since the others cannot be solved in a reasonable time.

Solution stability is an important part of our approach. As we increase the number of scenarios, we can expect the solution to eventually stabilize as the simulated approximation of our discrete choice model becomes increasingly accurate. Our goal is to find the minimum number of scenarios needed to reach a stable solution. We do this by analyzing the convergence of the objective function value as the number of scenarios increase. To this end, we take a look at three of the five previously tested instances with 40 OD pairs, 50 candidate

Fig. 6. Stability (optimal value vs number of scenarios for instance 1)



arcs and 2 types of resources. Figures 6-8 show how the optimal solution values vary with respect to the number of scenarios for the three instances. As these plots show, past 40 scenarios, the optimal solution values seem relatively stable and oscillating only in relatively small intervals. We note that with only 1 scenario, optimal values are far from what they should be when the discrete choice model governing route choice is properly modeled with a higer number of scenarios. This highlights the importance of considering an adequate number of scenarios to achieve an accurate result. This is also reflected in the optimal solutions for variables y themselves, which stabilize past 70, 20 and 60 scenarios for each instance respectively. We note that finding an ideal number of scenarios is dependent on a number of factors such as the network topology, the discrete choice model parameters and the impact control resources have on the path choice of users. As such, conducting a solution analysis is important in order to guarantee the quality of the optimal solution found.

With an adequate number of scenarios estimated at 40, we proceed to evaluate how increasing the number of OD pairs and candidate arcs affects the solution times. We also study the effects of changing the number of scenarios. Because we are attempting to solve very large instances, the timeout is set at 10 hours for the following experiments. Tables 8-10 present the results over 5 instances for varying quantities of these three problem dimensions. In these tables is a new metric: cgap_f which is the gap in percentage between the value of the best integer solution, optimal or at timeout, and the best upper bound found. We note that when an instance times out, its run time is not considered for the average time, but the instance is included for the average gaps calculations. Table 8 shows the increase in solution times with respect to the number of scenarios. This is further illustrated in Figure 9

Fig. 7. Stability (optimal value vs number of scenarios for instance 2)

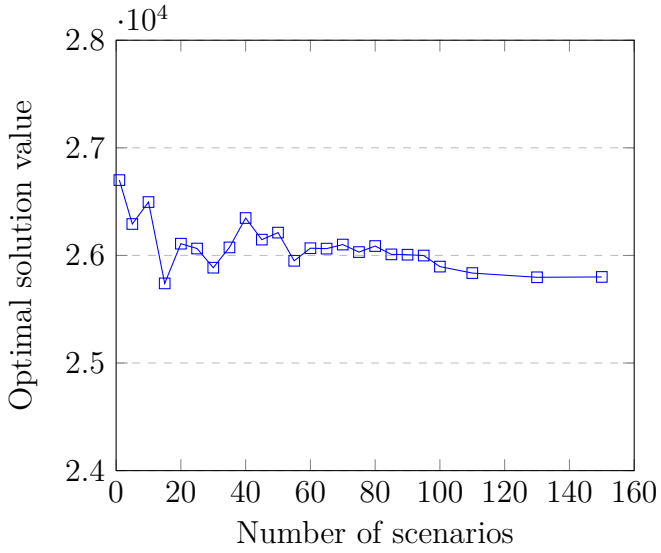
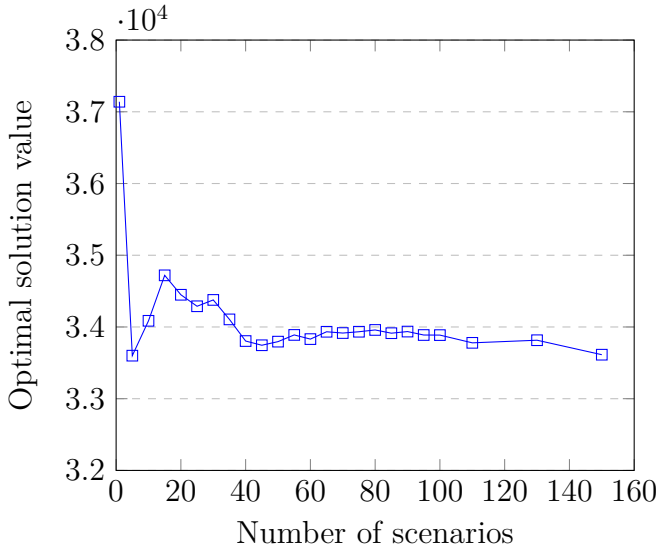


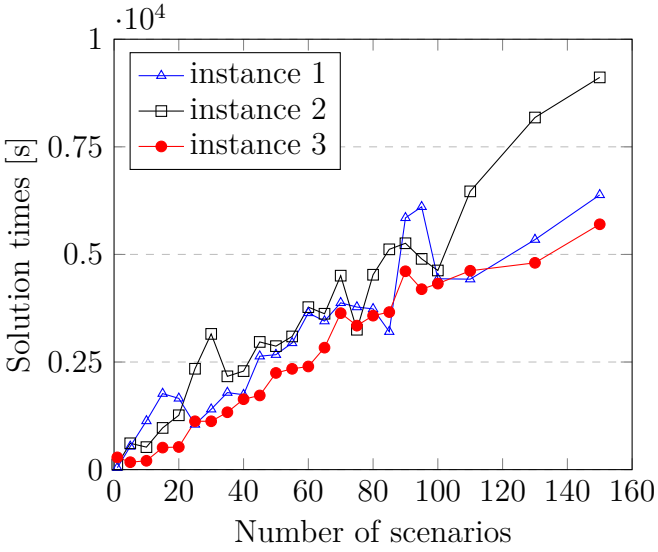
Fig. 8. Stability (optimal value vs number of scenarios for instance 3)



where the solution times for three of the instances have been individually plotted. As can be observed, the growth of the run times is fairly linear as a function of the number of scenarios,

When increasing the number of OD pairs, results are more contrasted, as can be seen in Table 9. Although the majority of the instances were solved to optimality within the time limit, we can observe that the growth in difficulty is not as smooth as with an increasing number of scenarios. The * denotes that the instance with 80 OD pairs that could not be solved caused a memory related error at approximately 32000 seconds (this time was not included in the average solution time). From the average cgap_f of 3.71%, it can be inferred

Fig. 9. Solution times vs number of scenarios



that this problematic instance was still at 18.54% when the process stopped. However, considering the other instances were solved reasonably efficiently, it can be seen as an outlier.

Finally, the results obtained when varying the number of candidate arcs are reported in Table 10. Instances remain manageable up to 75 candidate arcs, but instances with 100 candidate arcs could not be solved to optimality. The ** indicate that the program ran out of memory at approximately 25000 seconds for all five instances. However, cgap_f reports a 1.90% average final gap indicating that the quality of the solutions is relatively good.

$\lvert S\rvert$	1	25	50	75	100	150
Avg time [s]	195	1627	2533	3390	4467	7311
Solved	5/5	5/5	5/5	5/5	5/5	5/5
gap_h [%]	1.83	0.05	0.02	0.03	0.00	0.11
gap_r [%]	23.68	8.64	8.65	7.59	6.60	5.68
cgap_f [%]	0.0	0.0	0.0	0.0	0.0	0.0

Table 8. Average results (5 instances with 50 candidate arcs and 40 OD pairs): varying the number of scenarios

7. Conclusion and Future Work

We introduced an FCP that integrates RUM models, thus allowing a stochastic path choice representation and different types of user behaviors. We developed several MILP reformulations of the bilevel programming model for our FCP. From one of these reformulations, based on Lagragian relaxation, we derived a Benders decomposition method, which allowed us to solve large instances based on a network for the city of Winnipeg.

$\|K\|$	20	40	60	80
Avg time [s]	1357	4727	5384	9821
Solved	5/5	5/5	5/5	4/5*
gap_h	0.00	0.00	0.55	0.21
gap_r	6.65	9.73	18.76	22.21
cgap_f	0.00	0.00	0.00	3.71

Table 9. Average results (5 instances with 50 candidate arcs and 40 scenarios): varying the number of OD pairs

$\sum_{a\in A} \min_{r\in R}\{\sigma_{ar}\}$	25	50	75	100
Avg time [s]	469	2528	6652	25000**
Solved	5/5	5/5	5/5	0/5
gap_h	0	0.01	0	0
gap_r	3.32	6.11	9.42	9.68
cgap_f	0.0	0.0	0.0	1.90

Table 10. Average results (5 instances with 40 OD pairs and 40 scenarios): varying the number of candidate arcs

There are many potential avenues of research following this work. One extension is the inclusion of arc capacities and congestion as it can be an important factor in transportation applications. In addition, there is potential in adapting our Benders decomposition method to other bilevel programming models where lower level problems reduce to shortest path computations, since this is a fairly common situation, for example, in network design problems.

Acknowledgements

This research was partially funded by the NSERC Discovery Grant program. We are grateful to Serge Bisaillon for his help on improving the code.

Third Article.

Benders Decomposition for a Class of Bilevel Programs with Applications to Network Design

by

Léonard Ryo Morin[1], Emma Frejinger[1], and Bernard Gendron[1]

([1]) Department of Computer Science and Operations Research and CIRRELT
Université de Montréal, Canada

This article will be submitted to a journal yet to be determined after adding numerical results on the hazmat problem.

My contributions for this paper include writing the mathematical developments for all three specific models, as well as the rest of the text. Bernard Gendron provided the insight leading to the general decomposition method as well as corrections to the mathematical details. Emma Frejinger helped with some useful comments and by helping rewrite certain sections.

Résumé. Dans cet article, nous présentons une nouvelle méthode de décomposition de Benders pouvant être appliquée à une forme particulière de modèle biniveau. L'élément clé de cette décomposition est la méthode utilisée pour transformer le modèle biniveau à un modèle à un seul niveau. Cela se fait en utilisant la dualité forte pour ajouter un terme à la fonction objectif du problème du suiveur qui garantit l'optimalité sous certaines conditions. Afin de montrer que cette forme et ces suppositions ne sont pas excessivement contraignantes, nous exposons le fait que cette forme contient une classe générique de problèmes biniveaux de design de réseaux. De plus, deux formulations de l'état de l'art pour des problèmes communs dans la littérature (design de réseau pour le transport de matières dangeureuses et capture de flot) sont adaptés à notre décomposition dans le but de démontrer sa polyvalence. Une troisième formulation, pour le problème de design et prix simultanés de réseau, est présenté à l'Annexe B.
Mots clés : programmation biniveau, design de réseau, décomposition de Benders, design de réseau pour le transport de matières dangeureuses, design et prix simultanés de réseau, capture de flot

Abstract. In this article, we present a novel Benders decomposition method that can be applied to a particular form of bilevel programs. The key element of this decomposition lies in the method used to transform the bilevel model into a single-level model. This is done by using strong duality to add a term to the objective function of the follower's problem which guarantees optimality under certain assumptions. To show that the particular form and assumptions required for this decomposition are not excessively restrictive, this form is shown to include a generic class of bilevel network design problems. Furthermore, two state of the art formulations for problems commonly found in the literature (hazmat transportation network design and flow capture) are adapted to our decomposition method to further demonstrate its versatility. A third formulation, for the joint network design and pricing problem, is presented in Appendix B.
Keywords: bilevel programming, network design, Benders decomposition, hazmat transportation network design, joint network design and pricing, flow capture

1. Introduction

The applications covered by bilevel programming typically have important consequences for the economy and its related fields [64, 21]. This is mainly due to their particular structure where one optimization problem contains another as a constraint. Concretely, this can be seen as a sequential game with a leader and a follower. The leader's problem contains the follower's problem, meaning that it makes its decisions while anticipating the reaction from the follower. The follower makes its decisions according to its own problem, which depends in some way on the leader's decisions.

Solving bilevel programs is known to be intrinsically difficult. One of the common difficulties is the intractability that often accompanies solving large instances [18]. This can be explained by the fact that even linear bilevel programs are $\mathscr{NP}$-hard [36]. Furthermore,

simply evaluating a solution for optimality is also $\mathscr{NP}$-hard [69]. Thus, finding ways to efficiently solve in practice these problems is important.

In this paper, we present a general Benders decomposition method that can be applied to problems of the form

$$Z = \min g(y) + (yF + e)x, \tag{1.1}$$

$$y \in Y, \tag{1.2}$$

$$x \in \arg\min_{x \in X}\{(yC + b)x\}, \tag{1.3}$$

where

- $y = (y^i)^{i=1,\ldots,m}$, $Y \subseteq \{0,1\}^m$, $Y \neq \emptyset$;
- $x = (x_j)_{j=1\ldots,n}$, $X = \{x \geq 0 | Ax \geq d\} \subseteq [0,u]^n$, $X \neq \emptyset$;
- C,F are $m \times n$ matrices and A is an $l \times n$ matrix;
- b,e are n-dimensional vectors and d is an l-dimensional vector.

We also make the following assumptions:

Assumption 5. *There exists an n-dimensional vector ω such that*

$$0 \leq (yF + e) \leq \omega, \ y \in Y.$$

Assumption 6. *There exists an n-dimensional vector Δ such that*

$$(yC + b) \geq \Delta > 0, \ y \in Y.$$

Before presenting our solution method, we mention that there are many approaches to solving bilevel programming models. We refer the reader to the works of [18] and [64] as they are two examples of papers providing overviews of these methods.

While adding linearized complementary slackness constraints and strong duality constraints are two popular methods to obtain a single-level formulation, the approach we take is slightly different. Instead of using strong duality to add a constraint equating the primal and dual objective functions of the follower's problem (the strong duality constraint), we add it to the objective function of the leader as a Lagrangian penalty. The approach that consists of penalizing the strong duality constraint is well-known in the literature on bilevel programming, where it is used in the context of iterative heuristic methods (see, e.g., [70, 43, 15, 11, 12]). It is on this particular one-level model that we apply our Benders decomposition method, which is a novel approach, to the best of our knowledge. Although there have been numerous works regarding the use of Benders decomposition in the context of bilevel programming, none of them is based on Lagrangian relaxation, like ours.

While (1.1)-(1.3) is of a particular form, it encompasses a number of problem classes. To illustrate this, we show how our decomposition can be applied to two distinct network design applications by reformulating corresponding state-of-the-art models. The two applications

explored are hazmat transportation network design and flow capture problems, which are special cases of bilevel multicommodity uncapacitated network design problems.

The remainder of the paper is structured as follows: Section 2 details the generalized decomposition method. Section 3 showcases properties in the general case of bilevel multicommodity uncapacitated network design problems. Sections 4 and 5 show how the proposed decomposition method applies to, respectively, the hazmat transportation network design and flow capture problems. Finally, Section 6 concludes our work with a brief summary and potential future research directions.

2. Benders Decomposition

The goal of this section is to demonstrate how our particular Benders decomposition applies to the bilevel programming model detailed in the introduction. We begin by reformulating (1.1)-(1.3) as a single-level model after which we expose a suitable structure to apply the decomposition itself. The dual of the follower's problem (1.3) can be written as

$$\max_{\pi \in \Pi} \pi d, \tag{2.1}$$

where $\pi = (\pi^k)^{k=1,\dots,l}$ is the l-dimensional vector of dual variables associated with constraints $Ax \geq d$ and $\Pi = \{\pi \geq 0 | \pi A \leq yC + b\}$. To replace the follower's problem, we first add the dual feasibility constraints $\pi \in \Pi$ to our model. We then need to replace the objective of the follower's problem either by complementary slackness conditions or by the strong duality constraint, which states that the objective values of the primal and the dual of the follower's problem are equal: $(yC + b)x = \pi d$. In our case, we add the strong duality constraint as a Lagrangian penalty in the objective function. This results in the following single-level Lagrangian relaxation of the original bilevel program:

$$Z(\lambda) = \min_{y \in Y, x \in X, \pi \in \Pi} g(y) + (yF + e)x + \lambda((yC + b)x - \pi d). \tag{2.2}$$

For this relaxation to be a reformulation of the problem defined by (1.1)-(1.3), we must demonstrate that λ can take a finite value which prioritizes the optimality of the follower's part of the objective function. To do this, we rewrite (2.2) as follows, noting that the variables π are not linked anymore with the variables x in the Lagrangian relaxation:

$$Z(\lambda) = \min_{y \in Y} g(y) + \left\{ \min_{x \in X} \{(yF + e)x + \lambda(yC + b)x\} - \lambda \max_{\pi \in \Pi} \pi d \right\} \tag{2.3}$$

$$= \min_{y \in Y} g(y) + \left\{ \min_{x \in X} (yF + e + \lambda(yC + b))x - \lambda \min_{x' \in X} (yC + b)x' \right\}. \tag{2.4}$$

Recalling our assumptions (5) and (6), we can see that, by setting $\lambda > \omega u / \min_{j=1\dots,n}\{\Delta_j\}$, we prioritize the optimality of the follower's problem over the

"x" part of the leader's objective function, since

$$\begin{aligned}\lambda &> \omega u / \min_{j=1\ldots,n}\{\Delta_j\} \\ &\geq (yF+e)x/(yC+b)_j,\ j=1,\ldots,n,\end{aligned}$$

which implies $\lambda(yC+b)_j > (yF+e)x, j=1,\ldots,n$. So, for any $y \in Y$, if x^* is an optimal solution to $\min_{x\in X}\{(yF+e+\lambda(yC+b))x\}$, then it is also an optimal solution to $\min_{x'\in X}\{(yC+b)x'\}$. This implies that the two minimization problems over X in (2.4) have the same optimal solution x^*, so for any $y \in Y$, we have

$$\min_{x\in X}\{(yF+e+\lambda(yC+b))x\} - \lambda \min_{x'\in X}\{(yC+b)x'\} = (yF+e)x^*.$$

In particular, if ($\overline{y}$,$\overline{x}$) is an optimal solution to the bilevel program (1.1)-(1.3), then ($\overline{y}$,x^*) is also an optimal solution and (2.2) is a one-level reformulation of this bilevel program for any $\lambda > \omega u / \min_{j=1\ldots,n}\{\Delta_j\}$.

This reformulation now lends itself to a Benders decomposition where the two inner minimization problems can be treated as subproblems in order to add cuts in the master problem defined over binary y variables. Optimality cuts for $\min_{x'\in X}(yC+b)x'$, which is already in its dual form, are easy to derive because X is independent of variables y and thus are simply given by

$$w_\pi \leq (yC+b)\widetilde{x}$$

for every extreme point $\widetilde{x}$ of X, where w_π is the variable that approximates the value of $\max_{\pi\in\Pi}\pi d = \min_{x'\in X}(yC+b)x'$. Feasibility cuts are unnecessary because $X \subseteq [0,u]^n$ is bounded.

Finding the expression of the cuts on $\min_{x\in X}(yF+e+\lambda(yC+b))x$ first requires us to linearize the yx variable product found in this objective function. This can be done by defining $v^i_j = y^i x_j, i=1,\ldots,n, j=1,\ldots,m$, which yields the following linear reformulation of $\min_{x\in X}(yF+e+\lambda(yC+b))x$:

$$\min_{x\in X, v\geq 0}(f+\lambda c)v + (e+\lambda b)x, \tag{2.5}$$

$$v^i_j \leq x_j, i=1,\ldots,n, j=1\ldots,m, \quad (\gamma^i_j \geq 0) \tag{2.6}$$

$$v^i_j \leq u y^i, i=1,\ldots,n, j=1\ldots,m, \quad (\gamma^{yi}_j \geq 0) \tag{2.7}$$

$$v^i_j \geq x_j - u(1-y^i),\ i=1,\ldots,n, j=1\ldots,m, \quad (\gamma^{(1-y)i}_j \geq 0) \tag{2.8}$$

where

- $v = (v^i_j)^{i=1,\ldots,m}_{j=1,\ldots,n}$ is an $m \times n$-dimensional vector;
- $f = (f^i_j)^{i=1,\ldots,m}_{j=1,\ldots,n}$ is an $m \times n$-dimensional vector such that $f^i_j = (F)_{ij}, i=1,\ldots,n, j=1\ldots,m$;

- $c = (c_j^i)_{j=1,\ldots,n}^{i=1,\ldots,m}$ is an $m \times n$-dimensional vector such that $c_j^i = (C)_{ij}, i = 1,\ldots,n, j = 1\ldots,m$;

We can then write the dual of this subproblem in order to derive the expression of the associated optimality cuts (because X does not depend on y, feasibility cuts are not needed):

$$\max \theta d - u \sum_{j=1}^{n} \{\gamma_j y + \gamma_j^{(1-y)}(\mathbb{1} - y)\} \tag{2.9}$$

$$\theta A + \sum_{i=1}^{m} \{\gamma^i - \gamma^{(1-y)i}\} \leq e + \lambda b, \tag{2.10}$$

$$-\gamma - \gamma^y + \gamma^{(1-y)} \leq f + \lambda c, \tag{2.11}$$

$$\theta \geq 0,\ \gamma, \gamma^y, \gamma^{(1-y)} \geq 0, \tag{2.12}$$

where

- θ is the l-dimensional vector of dual variables associated with constraints $Ax \geq d$;
- $\gamma = (\gamma_j^i)_{j=1,\ldots,n}^{i=1,\ldots,m}$ is an $m \times n$-dimensional vector of dual variables associated with constraints (2.6) (γ_j and γ^i are, respectively, the m-dimensional and the n-dimensional subvectors derived from γ);
- $\gamma^y = (\gamma_j^{yi})_{j=1,\ldots,n}^{i=1,\ldots,m}$ is an $m \times n$-dimensional vector of dual variables associated with constraints (2.7);
- $\gamma^{(1-y)} = (\gamma_j^{(1-y)i})_{j=1,\ldots,n}^{i=1,\ldots,m}$ is an $m \times n$-dimensional vector of dual variables associated with constraints (2.8) ($\gamma_j^{(1-y)}$ and $\gamma^{(1-y)i}$ are, respectively, the m-dimensional and the n-dimensional subvectors derived from γ);
- $\mathbb{1}$ is the m-dimensional vector of 1's.

The expression of the optimality cuts is as follows, where w_x is the variable that approximates the value of (2.5)-(2.8):

$$w_x \geq \widetilde{\theta} d - u \sum_{j=1}^{n} \{\widetilde{\gamma}_j y + \widetilde{\gamma}_j^{(1-y)}(\mathbb{1} - y)\}$$

for every extreme point $(\widetilde{\theta}, \widetilde{\gamma}, \widetilde{\gamma^y}, \widetilde{\gamma}^{(1-y)})$ of the polyhedron D defined by (2.10)-(2.12). By solving complementary slackness equations and feasibility conditions, we have the following values for $(\widetilde{\theta}, \widetilde{\gamma}, \widetilde{\gamma^y}, \widetilde{\gamma}^{(1-y)})$ if y variables take on integer values in the current master problem solution:

$$\widetilde{\theta} \text{ solves the dual of } \min_{x \in X}(yF + e + \lambda(yC + b))x; \tag{2.13}$$

$$\widetilde{\gamma}_j^i = \max\{0, -(f_j^i + \lambda c_j^i)\} y^i, i = 1,\ldots,n, j = 1\ldots,m, \tag{2.14}$$

$$\widetilde{\gamma}_j^{yi} = \max\{0, -(f_j^i + \lambda c_j^i)\}(1 - y^i), i = 1,\ldots,n, j = 1\ldots,m, \tag{2.15}$$

$$\widetilde{\gamma}_j^{(1-y)i} = \max\{0,(f_j^i + \lambda c_j^i)\} y^i, i = 1,\ldots,n, j = 1\ldots,m. \tag{2.16}$$

This important result means that, in this case, the cuts (v and w) on both subproblems can be generated by solving only $\min_{x \in X}(yF+e+\lambda(yC+b))x$. If this problem can be solved by a specialized algorithm, generating the cuts can become very efficient. However, in the case where binary variables y take on fractional values, which might happen if we solve the LP relaxation by Benders decomposition, the linear program (2.9)-(2.12) has to be solved in order the generate the optimality cut. We can speed up this process by setting the initial values of the variables using (2.13)-(2.16) evaluated with the current fractional solution y.

To summarize, we can write the Benders reformulation as:

$$Z = \min_{y \in Y, w_x, w_\pi} g(y) + w_x - \lambda w_\pi, \tag{2.17}$$

$$w_x \geq \widetilde{\theta} d - u \sum_{j=1}^{n} \{\widetilde{\gamma}_j y + \widetilde{\gamma}_j^{(1-y)}(\not\Vdash - y)\}, \quad (\widetilde{\theta}, \widetilde{\gamma}, \widetilde{\gamma}^y, \widetilde{\gamma}^{(1-y)}) \in ext(D), \tag{2.18}$$

$$w_\pi \leq (yC + b)\widetilde{x}, \quad \widetilde{x} \in ext(X), \tag{2.19}$$

where $ext(P)$ is the set of extreme points of polyhedron P.

3. Bilevel Uncapacitated Network Design

We dedicate this section to adapting the developments of Section 2 to the context of bilevel multicommodity uncapacitated network design problems. The goal is to showcase important characteristics specific to this setting that allow for an efficient solution process using specialized algorithms (here, shortest path computations) instead of solving LPs. In addition, this class of problems illustrate the case where the Benders subproblems are separable (here, by commodities).

In a network $G = (N,A)$ with nodes N, arcs A and commodities K, we assume a problem context where we define binary design variables y_{ij} and flow variables x_{ij}^k for each arc $(i,j) \in A$ and commodity $k \in K$. Given a demand $d^k > 0$ between the origin $O(k)$ and the destination $D(k)$ of every commodity $k \in K$, we define costs for each arc $(i,j) \in A$ as $0 \leq f_{ij}y_{ij}+e_{ij} \leq \omega_{ij}$ and $c_{ij}y_{ij} + b_{ij} \geq \Delta_{ij} > 0$ at the upper and lower levels, respectively, of the following bilevel multicommodity uncapacitated network design problem:

$$Z = \min_{y \in Y, x^k \in X_*^k, k \in K} g(y) + \sum_{k \in K} (fy + e) d^k x^k \tag{3.1}$$

where X_*^k is the set of optimal solutions to the following shortest path problem

$$\min \sum_{(i,j) \in A} (c_{ij}y_{ij} + b_{ij}) x_{ij}^k, \tag{3.2}$$

$$\sum_{(i,j) \in A} x_{ij}^k - \sum_{(j,l) \in A} x_{jl}^k = \begin{cases} -1, & \text{if } j = O(k), \\ 1, & \text{if } j = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad j \in N, \tag{3.3}$$

$$0 \leq x_{ij}^k \leq 1, \quad (i,j) \in A. \tag{3.4}$$

The dual of this shortest path problem is one of maximizing the potentials at each node under the constraint that the difference of potentials between two nodes linked by an arc is capped by the cost associated to using that arc, which can be written more specifically:

$$\max \pi_{D(k)}^k, \tag{3.5}$$

$$\pi_j^k - \pi_i^k \leq c_{ij} y_{ij} + b_{ij}, \quad a = (i,j) \in A, \tag{3.6}$$

$$\pi_{O(k)}^k = 0, \tag{3.7}$$

where variables π_i^k represent the potential at node $i \in N$, i.e., the shortest path length from $O(k)$ to node $i \in N$. The strong duality constraints can then be written as:

$$\sum_{(i,j)\in A} (c_{ij} y_{ij} + b_{ij}) x_{ij}^k = \pi_{D(k)}^k, \quad k \in K. \tag{3.8}$$

Introducing Lagrange multipliers λ^k for each of these constraints, we can then derive the following single-level relaxation of the bilevel program (3.1):

$$Z(\lambda) = \min g(y) + \sum_{k\in K} \{(fy + e) d^k x^k + \lambda^k ((cy + b) x^k - \pi_{D(k)}^k)\} \tag{3.9}$$

$$y \in Y;\ x^k \in X^k, \pi^k \in \Pi^k, \quad k \in K, \tag{3.10}$$

where X^k is the polyhedron defined by (3.3)-(3.4) and Π^k is the polyhedron defined by (3.6)-(3.7). Given our assumptions on the costs, this Lagrangian relaxation becomes a reformulation of the bilevel program (3.1) whenever

$$\lambda^k > d^k \sum_{(i,i)\in A} \omega_{ij}/\Delta,$$

where $\Delta = \min_{(i,j)\in A}\{\Delta_{ij}\}$, since this implies that $\lambda^k(c_{ij}y_{ij} + b_{ij}) > (fy + e)d^k x^k$, $k \in K, (i,j) \in A$. Thus, we can set $\lambda^k = d^k \rho, k \in K$, where $\rho > \sum_{(i,i)\in A} \omega_{ij}/\Delta$.

It follows, with such a choice of λ, that the bilevel program (3.1) can be rewritten as:

$$Z = \min_{y\in Y} g(y) + \sum_{k\in K} d^k \left\{ \min_{x^k \in X^k} (fy + e + \rho(cy + b)) x^k - \rho \max_{\pi^k \in \Pi^k} \pi_{D(k)}^k \right\}. \tag{3.11}$$

This implies that the two inner optimization problems can be solved simultaneously by a shortest path algorithm with arc lengths $(fy + e + \rho(cy + b))_{ij}$. These problems can be decomposed by commodities $k \in K$ and can be solved by Dijkstra's algorithm, since the costs are positive. The algorithm can be executed only once per origin as long as it provides the shortest path to each other node in the network, thus providing the solution to every destination with the same origin. Therefore, there are at most $|N|$ single origin-multiple destinations shortest path subproblems to solve.

In regards to our more general Benders reformulation (2.17)-(2.19), in the case of a an integer solution y, this shortest path algorithm provides the cuts on w^k and v^k simultaneously,

since the shortest paths are the same and only the arc costs differ, where w_π^k and w_x^k are the variables that approximate the values of the subproblems $\max_{\pi^k \in \Pi^k} \pi^k_{D(k)}$ and $\min_{x^k \in X^k}(fy + e + \rho(cy + b))x^k$, respectively. More precisely, cuts on w_π^k are given by

$$w_\pi^k \leq \sum_{(i,j)\in A} (c_{ij}y_{ij} + b_{ij})\delta_{ij}^{kp}, \quad k \in K, p \in P^k, \tag{3.12}$$

where P^k is the set of simple paths from $O(k)$ to $D(k)$ for each $k \in K$ and $\delta_{ij}^{kp} = 1$, if arc $(i,j) \in A$ belongs to path $p \in P^k$ for $k \in K$, and 0, otherwise. This follows immediately from the fact that extreme points of X^k correspond to paths in P^k.

Cuts on w_x^k can be obtained by linearizing the products of variables $y_{ij}x_{ij}^k$ with variables v_{ij}^k, which yields the following linear reformulation of $\min_{x^k \in X^k}(fy + e + \rho(cy + b))x^k$:

$$\min_{x^k \in X^k, v^k \geq 0} (f + \rho c)v^k + (e + \rho b)x^k, \tag{3.13}$$

$$v^k \leq x^k, \quad (\gamma^k \geq 0) \tag{3.14}$$

$$v^k \leq y, \quad (\gamma^{yk} \geq 0) \tag{3.15}$$

$$v^k \geq x^k - (\mathbb{1} - y). \quad (\gamma^{(1-y)k} \geq 0) \tag{3.16}$$

The dual of this linear program is then:

$$\max \theta_{D(k)}^k - \gamma^{k(y)}y - \gamma^{k(1-y)}(\mathbb{1} - y), \tag{3.17}$$

$$\theta_j^k - \theta_i^k + \gamma^k - \gamma^{k(1-y)} \leq (e + \rho b), \tag{3.18}$$

$$-\gamma^k - \gamma^{k(y)} + \gamma_{ij}^{k(1-y)} \leq (f + \rho c), \tag{3.19}$$

$$\theta^k, \gamma^k, \gamma^{k(y)}, \gamma^{k(1-y)} \geq 0. \tag{3.20}$$

The Benders cuts on w_x^k are then expressed as:

$$w_x^k \geq \widetilde{\theta}_{D(k)}^k - \widetilde{\gamma}^{k(y)}y - \widetilde{\gamma}^{k(1-y)}(\mathbb{1} - y), \quad k \in K, (\widetilde{\theta}^k, \widetilde{\gamma}^k, \widetilde{\gamma}^{k(y)}, \widetilde{\gamma}^{k(1-y)}) \in ext(D^k), \tag{3.21}$$

where D^k is the polyhedron defined by (3.18)-(3.20). For any $y \in Y$, an optimal solution to this linear program is given by:

$$\widetilde{\theta}^k \text{ solves the dual of the shortest path problem } \min_{x^k \in X^k}(fy + e + \rho(cy + b))x^k; \tag{3.22}$$

$$\widetilde{\gamma}_{ij}^k = \max\{0, -(f_{ij} + \rho c_{ij})\}y_{ij}, \quad (i,j) \in A, \tag{3.23}$$

$$\widetilde{\gamma}_{ij}^{yk} = \max\{0, -(f_{ij} + \rho c_{ij})\}(1 - y_{ij}), \quad (i,j) \in A, \tag{3.24}$$

$$\widetilde{\gamma}_{ij}^{(1-y)k} = \max\{0,(f_{ij} + \rho c_{ij})\}y_{ij}, \quad (i,j) \in A. \tag{3.25}$$

If y takes on fractional values (derived from solving the LP relaxation by Benders decomposition), then (3.22)-(3.25) still provides a feasible solution, which can be used to warm start the solution of the linear program (3.17)-(3.20).

	$\overline{y} \in \{0,1\}$	$\overline{y} \in [0,1]$
w^k	Shortest paths	LP with warm start
Π^k		Shortest paths

Table 11. What needs to be solved for cuts on w and Π

Table 11 summarizes what needs to be solved to obtain the necessary cuts on w_x^k and w_π^k. In three cases out of four, shortest paths can be solved relatively quickly by Dijkstra's algorithm, but in the fractional case, cuts on w_x^k are obtained by solving an LP, which can be initialized with shortest path computations.

To summarize, we can write the Benders reformulation specialized to bilevel multicommodity uncapacitated network design problems as:

$$Z = \min_{y \in Y, w_x^k, w_\pi^k, k \in K} g(y) + \sum_{k \in K} d^k \{w_x^k - \rho w_\pi^k\} \tag{3.26}$$

subject to (3.12) and (3.21).

In the following sections, we demonstrate how two distinct applications can make use of the methods presented so far. We begin by introducing the nature of the problem and a brief literature review showcasing state of the art model formulations and solution methods. We then reveal how one of these models for each problem can be adapted to fit the required bilevel form to apply our decomposition method.

4. Hazmat Transportation Network Design

This section is devoted to the hazmat transportation network design problem. This problem is chosen as an example for our method. We begin by exploring the literature surrounding this problem and subsequently show how our decomposition method applies to one of the more recent formulations.

4.1. Literature Review

The HTNDP typically consists of a governing body setting regulations for users transporting hazardous materials within a network with the goal of minimizing the potential risk to the general population and the environment. This can be done with a combinatorial approach which corresponds to choosing on which links hazardous materials can be transported. Users then adapt their route choice based on which links are available and their own preferences.

The combinatorial approach takes root in the work of [40] where a bilevel model is proposed and subsequently converted into a single-level formulation through complementary slackness equations. [9] puts forward a different perspective where the route choice of carriers is not explicitly modelled. [30] proposes a cutting plane approach to deal with the hazmat

transportation network design problem. [2] provides a proof that the hazmat transportation network design problem is NP hard even for a single commodity. They also make use of strong duality instead of complementary slackness to transform the bilevel model of [40] to a single-level model. Furthermore, they ensure that when multiple shortest paths exist for the same commodity, the one with the most risk is selected. [23] uses the model from [2] as a starting point to propose a multi-cut Benders decomposition. They also include improvements in the form of Pareto-optimal cut generation and retaining the flow conservation constraints in the leader's problem in order to ensure that the subproblems, where shortest paths are found, are always feasible. To the best of our knowledge, the work of [23] is the only one proposing a Benders decomposition method and is thus, very relevant to our work. As they use the model of [2], the transition from the bilevel model to a single-level model is done through strong duality constraints and subsequently, they apply Benders decomposition. However, our approach differs in that we relax the strong duality constraints by adding them to the objective function with a weight large enough to ensure an exact reformulation as opposed to a relaxation. We then apply Benders decomposition to that model while exploiting the structure of the subproblems.

Aside from the body of work focused on a combinatorial approach to the bilevel hazmat network design problem, we quickly acknowledge the alternative perspective of toll-setting. [48] appears to be the first to introduce the idea of using toll setting in a hazmat transportation context arguing that it may be more realistic as governing authorities often do not have the right to close specific network links to hazmat transportation. [10] extend the work of [48] to take into account risk equity meaning that risk from the hazmat transportation is spread out more evenly as to not unfairly subject a certain part of the population to an excessive amount of risk.

4.2. Applying the Decomposition

We now turn our attention to the bilevel model found in [2] as it seems to be one of the more advanced and recent models to the best of our knowledge. The leader's problem consists of choosing which arcs $(i,j) = a \in A$ in network $G = (N,A)$ can be used for the transportation of hazardous materials by setting the values of binary variables y_{ij}. The objective is to minimize, for each OD pair $k \in K$, the risk r_{ij}^k scaled by the demand d^k which is fixed and known. The follower's problem is that of the users finding the shortest path, according to arc costs c_{ij}^k and their availability y_{ij}, from their origin $O(k)$ to their destination $D(k)$. The choice of the links forming this shortest path is represented by variables x_{ij}^k.

$$\min \sum_{k \in K} \sum_{(i,j) \in A} r_{ij}^k d^k x_{ij}^k, \tag{4.1}$$

$$y_{ij} \in \{0,1\}, \qquad \forall (i,j) \in A, \tag{4.2}$$

$$\min \sum_{k \in K} \Big(\sum_{(i,j) \in A} c_{ij}^k x_{ij}^k - \frac{1}{R} \sum_{(i,j) \in A} r_{ij}^k x_{ij}^k \Big), \tag{4.3}$$

$$\sum_{(i,j) \in A} x_{ij}^k - \sum_{(j,l) \in A} x_{jl}^k = \begin{cases} -1, & \text{if } j = O(k), \\ 1, & \text{if } j = D(k), \\ 0, & \text{otherwise,} \end{cases} \qquad \forall j \in N, k \in K, \tag{4.4}$$

$$x_{ij}^k \leq y_{ij}, \qquad \forall (i,j) \in A, k \in K, \tag{4.5}$$

$$x_{ij}^k \in \{0,1\}, \qquad \forall (i,j) \in A, k \in K. \tag{4.6}$$

The leader's objective function (4.1) consists of a minimization of the total risk multiplied by the demand across all arcs and OD pair. The follower's objective function (4.1) aims to minimize the sum of arc costs used for each OD pair. Of note is the term $-\frac{1}{R} \sum_{(i,j) \in A} r_{ij}^k x_{ij}^k$ where R is the sum over all OD pairs of the maximum risk path values. This term, under the assumption that c_{ij}^k is integer, guarantees that, amongst multiple shortest paths, the one with the maximum risk is selected. Constraints (4.4) are the classical flow conservation constraints and (4.5) are the network design constraints stating that an arc can only be used by the follower if it has been made available by the leader.

To be able to apply our Benders decomposition method to this model, we must first tackle two issues. The first is to reformulate the follower's problem to remove constraints (4.5) in order for it to have a feasible space independent of variables y_{ij}. We achieve this by adding them to both objective functions (4.1) and (4.3) with a large enough penalty M, for example $M \geq \max_{k \in K} \sum_{(i,j) \in A} c_{ij}^k$. The second issue concerns the integrality constraint on variables x_{ij}^k (4.6). However, they can simply be relaxed due to the total unimodularity of the follower's problem. Lastly, we decompose the problem by commodities k. We thus rewrite the follower's problem as the following $|K|$ problems:

$$\min \sum_{(i,j) \in A} (c_{ij}^k - \frac{r_{ij}^k}{R} + M(1 - y_{ij})) x_{ij}^k, \tag{4.7}$$

$$\sum_{(i,j) \in A} x_{ij}^k - \sum_{(j,l) \in A} x_{jl}^k = \begin{cases} -1, & \text{if } j = O(k), \\ 1, & \text{if } j = D(k), \\ 0, & \text{otherwise,} \end{cases} \qquad \forall j \in N, \tag{4.8}$$

$$0 \leq x_{ij}^k \leq 1, \qquad \forall (i,j) \in A. \tag{4.9}$$

With the appropriate mathematical program form, we now only need to verify our Assumptions (5)-(6). Assumption (5) is verified because the term in x of (4.1) can only take positive values in a minimization problem. Assumption (6) holds since the only negative term (with parameters r_{ij}^k) is by definition ≤ 1 and path costs c_{ij}^k are assumed to be integer and > 0.

By applying our methodology developed in Sections 2 and 3, we have to solve two shortest path problems, for each $k \in K$, with the following arc costs in order to generate cuts on w and Π respectively with a current integer solution $\overline{y}$:

$$\lambda^k\left(c_{ij}^k - \frac{r_{ij}^k}{R} + M(1-y_{ij})\right) + r_{ij}^k d^k + M(1-y_{ij}) \tag{4.10}$$

and

$$c_{ij}^k - \frac{r_{ij}^k}{R} + M(1-y_{ij}). \tag{4.11}$$

We have the following $|K|$ problems to solve for cuts on w_x^k with a fractional y:

$$\max \pi_{D(k)} - \sum_{(i,j)\in A} \gamma_{ij}^{k(y)} y_{ij} - \sum_{(i,j)\in A} \gamma_{ij}^{k(1-y)}(1-y_{ij}), \tag{4.12}$$

$$\pi_j^k - \pi_i^k + \gamma_{ij}^k - \gamma_{ij}^{k(1-y)} \leq r_{ij}^k d^k + M + (\lambda^k c_{ij}^k - \frac{\lambda^k}{R} r_{ij}^k + \lambda^k M), \qquad \forall (i,j) \in A, \tag{4.13}$$

$$-\gamma_{ij}^k - \gamma_{ij}^{k(y)} + \gamma_{ij}^{k(1-y)} \leq -\lambda^k M - M, \qquad \forall (i,j) \in A, \tag{4.14}$$

$$\gamma_{ij}^k, \gamma_{ij}^{k(y)}, \gamma_{ij}^{k(1-y)} \geq 0, \qquad \forall (i,j) \in A. \tag{4.15}$$

Finally, we can write our Benders reformulation for the hazmat transportation network design problem as:

$$\min \sum_{k\in K} (w_x^k - \lambda^k w_\pi^k) \tag{4.16}$$

subject to (4.2) and

$$w_x^k \leq \pi_{D(k)} - \sum_{(i,j)\in A} \gamma_{ij}^{k(y)} y_{ij} - \sum_{(i,j)\in A} \gamma_{ij}^{k(1-y)}(1-y_{ij}) \tag{4.17}$$

$$\forall k \in K, (\pi^k, \gamma^k, \gamma^{k(y)}, \gamma^{k(1-y)}) \in ext(D^k),$$

$$w_\pi^k \geq \sum_{(i,j)\in A} (c_{ij}^k - \frac{r_{ij}^k}{R} + M(1-y_{ij}))\delta_{ij}^{kp}, \qquad \forall k \in K, p \in P^k, \tag{4.18}$$

where D^k is the polyhedron defined by (4.12)-(4.15).

4.3. Connectivity Cuts

In this Section, we present the idea of connectivity cuts in general and how they are implemented in our solution method. Many $\overline{y}$ solutions proposed by the master problem are unlikely to form paths between the OD pairs which can lead to a significant waste of time. It is therefore advantageous to add connectivity cuts before and during the solution process.

To find a general expression for the connectivity cuts, we define two proper subsets of N. The first, S^k, must contain $O(k)$ and must not contain $D(k)$. The second is simply defined as $\overline{S}^k = N \setminus S^k$. We note that the definition of these two sets corresponds to s-t cuts in graph theory. We can now propose the following expression:

$$\sum_{(i,j)\in(S^k,\overline{S}^k)} y_{ij} \geq 1, \qquad k \in K, S^k \subset N. \tag{4.19}$$

This generic constraint states that at least one arc (i,j) in the cut set defined by S^k and $\overline{S}^k$ has to be opened. Two particular cases of (4.19) are the resulting cuts for origins and destinations of all OD pairs. They can be expressed as

$$\sum_{(i,j)\in A^k} y_{ij} \geq 1, \qquad k \in K, i = O(k) \tag{4.20}$$

and

$$\sum_{(i,j)\in A^k} y_{ij} \geq 1, \qquad k \in K, j = D(k). \tag{4.21}$$

Furthermore, we can define similar cuts for transfer nodes as

$$y_{ij} \leq \sum_{(j,l)\in A^k} y_{jl}, \qquad j \in T, \forall(i,j) \tag{4.22}$$

and

$$\sum_{(i,j)\in A^k} y_{ij} \geq y_{jl}, \qquad j \in T, \forall(j,l) \tag{4.23}$$

where $T = \{j \in N | \nexists k$ such that $j = O(k)$ or $j = D(k)$. Constraints (4.20)-(4.23) are similar in behavior to flow conservation constraints. They ensure that if an arc exiting a node is open, then there must be at least one arc entering that node that is open and vice-versa.

The final method of generating connectivity cuts is based on the $\overline{y}$ given by the master problem at any given iteration of the solution process. We first define the support graph $G(\overline{y}) = (N,A(\overline{y}))$ where $A(\overline{y}) = \{(i,j) \in A | \overline{y}_{ij} > 0\}$. Then, starting from $O(k)$, for every OD pair k, $G(\overline{y})$ is traversed forwards. If $D(k)$ cannot be reached, then S^k in equation (4.19) is defined as the set of nodes successfully visited by the search. This type of constraint is added during the solution process while constraints (4.20)-(4.23) are added to the model before the optimization.

4.4. Heuristic

We propose a simple heuristic to provide our model with initial cuts on variables w^k and Π^k. The procedure can be divided in two parts: the first consists of finding values for variables y_{ij} and the second consists of generating cuts based on those values.

Determining which arcs to open is done by solving shortest path problems with regards to arc costs c^k_{ij}. A candidate solution $\overline{y}_{ij}$ is built by simply setting y_{ij} to 1 if the corresponding arc (i,j) is part of any of the $|K|$ shortest paths. In order to find more solutions, this process is repeated S times where the arc costs of the next iteration are given by

$$c_{ij}^{k(s+1)} = \begin{cases} c_{ij}^{k(s)} * Q^c, & \text{if } y_{ij}^{(s)} = 1, \\ c_{ij}^{k(s)}, & \text{otherwise,} \end{cases} \tag{4.24}$$

where Q^c is a number strictly greater than 1. This penalizes arcs that are repeatedly found to be part of the shortest paths for each OD pair which encourages new solutions to be found.

We also build a second set of solutions for variables y_{ij} by using the same approach of solving shortest path problems multiple times, but with regards to arc risks r_{ij}^k. Again, y_{ij} is set to 1 if the corresponding arc is part of any of the shortest paths found and the process is repeated S times with a multiplier Q^r applied to the risk $r_{ij}^{k(s)}$ of opened arcs of the previous iteration.

Finally, as every $\overline{y}$ found is integer, we can generate the cuts on w^k and Π^k by solving shortest path problems with regards to (4.10) and (4.11) respectively, as detailed in Section 4.2.

5. Flow Capture

In this Section, we look at the flow capture problem. We judge this problem to be of interest because it can be used for many applications related to providing a service to traffic flows within a network or intercepting unlawful drivers and vehicles. Following the same structure as Section 4, the surrounding literature is reviewed and our method is applied to a recent formulation.

5.1. Literature Review

Flow capture problems differ from the previous two applications as it does not involve network design decisions. Instead, the leader's problem consists of installing flow capturing resources on certain arcs of the network with the goal of intercepting the most traffic flow. The follower's problem consists of finding the shortest path according to the users' preferences in regards to the base network characteristics and the presence of flow capturing resources. The user's response to these resources can be evasive, cooperative or neutral depending if the resource in question raises, lowers or does not affect the cost of using that arc respectively.

Flow capture problems find their origin in the works of [33] and [8] and have since featured many variations with various characteristics. [74] provide an overview of these features and present a general formulation which can be adapted to implement them. We note, however, that most works in the field, for example [42] and [50], do not use bilevel models to formulate the cyclic interaction between the installation of resources to capture flow and the users' potential change of route choice in response. Instead, a maximum deviation users are willing to make is calculated in order to define the set of all possible paths for a given OD pair. One

of the works using a bilevel model is that of [4] where the follower's problem is a shortest path problem, but behaviorally it is identical to the model in [50] on which it is based.

5.2. Applying our Decomposition

The model which serves as a basis for our decomposition model is the one we proposed in this thesis. This model is characterized by its ability to represent various types of behaviors exhibited by the users in regards to various types of resources. More importantly, this is done through discrete choice model simulation in an arc-based shortest path context rather than a pre-calculated path set. We note that the version presented here is restricted to deterministic path choice and a single user category. For the complete model and our Benders reformulation of it, we refer the reader to our work presented in the preceding chapter.

Again, we denote the graph as $G = (N,A)$ and the demand as d^k for each OD pair $k \in K$. The cost of installing resource of type r on an arc $(i,j) \in A$ is indicated by c_{ij}^r and its capture percentage is indicated by q_{ij}^r. Binary variables y_{ij}^r are equal to 1 if a resource of type r is installed on arc (i,j) and 0 otherwise. The additional arc cost associated with the presence of a resource is denoted by β_{ij}^r while the base arc cost associated with its characteristics is denoted by ε_{ij}. Finally, variables x_{ij}^k are the flow variables.

$$Z = \max \sum_{k \in K} \sum_{(i,j) \in A} \sum_{r \in R_{ij}} (q_{ij}^r d^k) y_{ij}^r x_{ij}^k, \tag{5.1}$$

$$\sum_{(i,j) \in A} \sum_{r \in R_{ij}} c_{ij}^r y_{ij}^r \leq b, \tag{5.2}$$

$$\sum_{r \in R_{ij}} y_{ij}^r \leq 1, \quad (i,j) \in A, \tag{5.3}$$

$$y_{ij}^r \in \{0,1\}, \quad (i,j) \in A, r \in R_{ij}, \tag{5.4}$$

$$\min \sum_{k \in K} \sum_{(i,j) \in A} \sum_{r \in R_{ij}} (\beta_{ij}^r y_{ij}^r + \varepsilon_{ij}) x_{ij}^k, \tag{5.5}$$

$$\sum_{(i,j) \in A} x_{ij}^k - \sum_{(j,l) \in A} x_{jl}^k = \begin{cases} -1, & \text{if } j = O(k), \\ 1, & \text{if } j = D(k), \\ 0, & \text{otherwise,} \end{cases} \qquad \forall j \in N, \tag{5.6}$$

$$x_{ij}^k \geq 0, \quad (i,j) \in A, k \in K. \tag{5.7}$$

The objective function of the leader (5.1) consists of maximizing the flow captured over all OD pairs. Constraint (5.2) is a budget constraint limiting the number of resources which can be installed. Constraints (5.3) limits the number of resources installed per arc to 1. Equations (5.5)-(5.7) consists of a shortest path problem for all OD pairs where the cost of using an arc takes into account the presence of a resource.

The bilevel model matches (1.1)-(1.3) and therefore only (5)-(6) need to be validated before applying our decomposition method. Assumption (5) holds as the number of arcs,

potential installed resources and demand all take finite values so an upper bound can be calculated. Assumption (6) is valid because the follower's problem is a shortest path problem with arc costs > 0. We note that this problem can also be decomposed by commodity k.

Two shortest path problems need to be solved for each $k \in K$ with the following arc costs in order to generate cuts on w and Π respectively with a current integer solution $\overline{y}$:

$$\sum_{r \in R_{ij}} \lambda^k (\beta_{ij}^r y_{ij}^r + \varepsilon_{ij}) - q_{ij}^r y_{ij}^r \tag{5.8}$$

and

$$\sum_{r \in R_{ij}} \beta_{ij}^r y_{ij}^r + \varepsilon_{ij}. \tag{5.9}$$

We have the following $|K|$ problems to solve for cuts on w with a fractional $\overline{y}$:

$$\min -\pi_{D(k)}^k + \sum_{(i,j) \in A} \sum_{r \in R_{ij}} \left(\gamma_{ij}^{rk(y)} y_{ij}^r + \gamma_{ij}^{rk(1-y)} (1 - y_{ij}^r) \right), \tag{5.10}$$

$$\pi_i^k - \pi_j^k - \sum_{r \in R_{ij}} (\gamma_{ij}^{rk} - \gamma_{ij}^{rk(1-y)}) \geq -\lambda^k \varepsilon_{ij}, \quad (i,j) \in A, \tag{5.11}$$

$$\gamma_{ij}^{rk} + \gamma_{ij}^{rk(y)} - \gamma_{ij}^{rk(1-y)} \geq q_{ij}^r - \omega \beta_{ij}^r, \quad (i,j) \in A, r \in R_{ij}, \tag{5.12}$$

$$\pi_{O(k)}^k = 0, \tag{5.13}$$

$$\gamma_{ij}^{rk}, \gamma_{ij}^{rk(y)}, \gamma_{ij}^{rk(1-y)} \geq 0, \quad (i,j) \in A, r \in R_{ij}. \tag{5.14}$$

Finally, we can write our Benders reformulation for the hazmat transportation network design problem as:

$$\max \sum_{k \in K} d^k (w^k + \lambda^k \Pi^k) \tag{5.15}$$

subject to (5.2)-(5.4) and

$$w^k \leq -\pi_{D(k)} + \sum_{(i,j) \in A} \sum_{r \in R_{ij}} \left(\gamma_{ij}^{rk(y)} y_{ij}^r + \gamma_{ij}^{rk(1-y)} (1 - y_{ij}^r) \right), \tag{5.16}$$

$$k \in K, (\pi, \gamma, \gamma^y, \gamma^{(1-y)}) \in ext(\mathscr{D}_1),$$

$$\Pi^k \leq \sum_{(i,j) \in A} \sum_{r \in R_{ij}} (\beta_{ij}^r y_{ij}^r + \varepsilon_{ij}) x_{ij}^k, \quad k \in K, p \in ext(\mathscr{D}_2), \tag{5.17}$$

where $\mathscr{D}_1^k$ is the solution space of (5.10)-(5.14) and $\mathscr{D}_2^k$ is the solution space of (5.5)-(5.7).

6. Conclusion and Future Work

In this paper, we presented a novel Benders decomposition method for bilevel models of a certain form. In particular, we showed how it can be applied to models with a multi-commodity shortest path problem as the follower's problem. We also demonstrated that it

is possible to reformulate models for common applications in order to match the required form for this decomposition method.

A clear future endeavor to pursue would be to conduct computational experiments comparing the models from the literature and the models obtained from applying our decomposition, in particular, for the case of the HTNDP.

Acknowledgements

This research was partially funded by the NSERC Discovery Grant program.

Conclusion

This thesis contributed to the state of the art of traffic prediction and bilevel network design. The contribution took the form of three articles. The first of these articles was focused on two aspects: the analysis of GPS data and estimating a recursive logit model. The data analysis included is only a fraction of a much larger project conducted for CargoM, but it demonstrates the wide range of information that can be extracted from it. In fact, this information fuels the second part of the article as the observed paths taken by the vehicles are used to estimate the parameter values of a recursive logit model. The calibrated model can subsequently explain the behavior of drivers as a function of the model parameters.

The second article addressed the issues surrounding the integration of discrete choice models, such as the recursive logit model used in the first article, in the flow capture problem. The flow capture problem was an obvious candidate for this work because of its many applications and the important role that the demand plays in it. The first point of interest in this article is how we adapt the simulation approach of [56] to the more challenging context of network optimization. Expressing the RUM model in arc utility space instead of arc probability space leads to linear constraints instead of non-linear ones. By creating enough scenarios based on realizations of the random term part of the arc utility definition, we have an estimation of a discrete choice model. This estimation becomes more accurate as the number of scenarios increases. However, the simulation approach as a whole creates a large set of variables and constraints. The second point of interest is the Benders decomposition method applied to our flow capture problem formulation. This decomposition features cuts that can often be calculated by a shortest path algorithm rather than solving a linear program. As a result, relatively large instances can be solved in reasonable times.

The third article demonstrated that the decomposition method developed in the second article can be generalized to a particular form of bilevel models. The first part of this article is based on generic bilevel formulations and how our method can applied to them while exploiting their structure. The second part of the article consists of adapting three different state of the art models to the required bilevel form. The three applications explored are: the hazmat transportation network design problem, the joint network design and pricing problem, and the flow capture problem. We showed that the particular form required for

the decomposition does in fact encompass many formulations for problems commonly found in the literature.

As a whole, the three articles provided the means of integrating better demand models in problems with bilevel formulations. We began with the estimation of a discrete choice model based on real GPS data. We then showed how it can be included in a general flow capture problem formulation. Finally, we demonstrated how the decomposition method used in the flow capture problem could be used for many more applications.

Limitations and Outlook

In this final section, we take a look at the limitations we encountered throughout this thesis. We also discuss how these limitations can pave the way for future research.

In the first article, there are many directions for potential additional endeavors. One of the main limitations of our work is the poor quality of the network data used for the map matching. After carefully inspecting it in a geographical information system application (QGIS), we noticed that some nodes, or intersections, are seemingly duplicated and separated by infinitesimal distances which caused continuity problems in the map matching process. Also, arcs appeared to be missing from certain intersections rendering some turns impossible. These issues limited the number of arcs in the observed paths. With a better representation of the network, we could have benefited from a larger number of observations for the recursive logit model parameter estimation. Furthermore, if the network data were to contain additional link attribute information, additional corresponding parameters could be estimated.

We should also mention that more descriptive analysis could be done. The focus was on the port of Montreal, but the GPS traces cover a much larger area including some interesting locations such as the rail yards where many stops are made. Additionally, this is a good reason to analyze vehicle tours which can provide further insights into potential policies or changes to the road network. However, we note that the size of the GPS data does not constitute a representative sample and thus expanding the dataset would be crucial in any further analysis.

The second article features numerous avenues of research. While reviewing the literature surrounding the FCP and other network optimization problems, we notice a number of modeling hypotheses which could be interesting to pursue. The first of these would be arc capacities. In our case, the network is supposed to be free of congestion, however modeling this phenomenon could be relevant in certain applications. However, typical capacity constraints would be problematic as they involve sums of flows set to be lesser than or equal to a maximum capacity. This would prevent us from directly applying our current decomposition.

In a similar vein, another compelling prospect would be to add facility capacities to cap the quantity of demand they can each serve. We could also consider rewarding earlier flow

capture by making the demand dependent on the arcs and setting it to a higher value for arcs closer to the origin of any given OD pair. Finally, a formulation where flows can only be captured once per OD pair could also be interesting to explore.

There are also some more practical possible improvements. A relatively high number of scenarios is necessary in order to achieve a good approximation of a discrete choice model. This can be problematic as more scenarios imply a larger model. As such, simulation techniques that reduce the variance of the random variable sampling are worth exploring as they have the potential to lower the number of scenarios required to reach a stable solution. Another interesting prospect would be to write a user cut callback function making use of parallelism. In the current implementation, this function, which derives cuts from fractional solutions, solves the appropriate LPs sequentially. However, with many scenarios, user categories and OD pairs, there is a significant gain in solving these problems in parallel.

For the third article, a computational comparison between state of the art models and the ones obtained without decomposition is the most important future work. This would allow the decomposition method to be further proven as an interesting solution method. The results from the second article show that it can indeed handle large instances, but being able to show similar results for other models would be a significant boon.

References

[1] A. R. Alho, L. You, F. Lu, L. Cheah, F. Zhao, and M. Ben-Akiva. Next-generation freight vehicle surveys: Supplementing truck gps tracking with a driver activity survey. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2974–2979. IEEE, 2018.

[2] E. Amaldi, M. Bruglieri, and B. Fortz. On the hazmat transport network design problem. In *International Conference on Network Optimization*, pages 327–338. Springer, 2011.

[3] U. Arıkan and S. D. Ahipasaoglu. On the existence and convergence of the markovian traffic equilibrium. 2017.

[4] O. Arslan, O. Jabali, and G. Laporte. Exact solution of the evasive flow capturing problem. *Operations Research*, 66(6):1625–1640, 2018.

[5] S. Bekhor, M. E. Ben-Akiva, and M. S. Ramming. Evaluation of choice set generation algorithms for route choice models. *Annals of Operations Research*, 144(1):235–247, 2006.

[6] M. Ben-Akiva and M. Bierlaire. Discrete choice methods and their applications to short term travel decisions. In *Handbook of transportation science*, pages 5–33. Springer, 1999.

[7] M. E. Ben-Akiva, T. Toledo, J. Santos, N. Cox, F. Zhao, Y. J. Lee, and V. Marzano. Freight data collection using gps and web-based surveys: Insights from us truck drivers' survey and perspectives for urban freight. *Case studies on transport policy*, 4(1):38–44, 2016.

[8] O. Berman, R. C. Larson, and N. Fouska. Optimal location of discretionary service facilities. *Transportation Science*, 26(3):201–211, 1992.

[9] L. Bianco, M. Caramia, and S. Giordani. A bilevel flow model for hazmat transportation network design. *Transportation Research Part C: Emerging Technologies*, 17(2):175–196, 2009.

[10] L. Bianco, M. Caramia, S. Giordani, and V. Piccialli. A game-theoretic approach for regulating hazmat transportation. *Transportation Science*, 50(2):424–438, 2015.

[11] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard. A bilevel model and solution algorithm for a freight tariff-setting problem. *Transportation Science*, 34:289–302, 2000.

[12] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard. A bilevel model for toll optimization on a multicommodity transportation network. *Transportation Science*, 35:345–358, 2001.

[13] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard. Joint design and pricing on a network. *Operations research*, 56(5):1104–1115, 2008.

[14] L. Brotcorne, P. Marcotte, G. Savard, and M. Wiart. Joint pricing and network capacity setting problem. *Advanced OR and AL Methods in Transportation (Jaszkiewicz, Kaczmarek, Zak and Kubiak, eds.). Publishing House of Poznan University of Technology*, 2005.

[15] M. Campelo, S. Dantas, and S. Scheimberg. A note on a penalty function approach for solving bilevel linear programs. *Journal of Global Optimization*, 16:245–255, 2000.

[16] E. Cascetta, A. Nuzzolo, F. Russo, and A. Vitetta. A modified logit route choice model overcoming path overlapping problems. specification and some calibration results for interurban networks. In *Transportation and Traffic Theory. Proceedings of The 13th International Symposium On Transportation And Traffic Theory, Lyon, France, 24-26 July 1996*, 1996.

[17] B. Colson, P. Marcotte, and G. Savard. Bilevel programming: A survey. *4or*, 3(2):87–107, 2005.

[18] B. Colson, P. Marcotte, and G. Savard. Bilevel programming: a survey. *4or*, 3(2):87–107, 2005.

[19] B. Colson, P. Marcotte, and G. Savard. An overview of bilevel optimization. *Annals of Operations Research*, 153(1):235–256, 2007.

[20] T. Dan and P. Marcotte. Competitive facility location with selfish users and queues. 2017.

[21] S. Dempe. *Foundations of bilevel programming*, volume 61. Springer Science & Business Media, 2002.

[22] M. Flaskou, M. A. Dulebenets, M. M. Golias, S. Mishra, and R. M. Rock. Analysis of freight corridors using gps data on trucks. *Transportation Research Record*, 2478(1):113–122, 2015.

[23] P. Fontaine and S. Minner. Benders decomposition for the hazmat transport network design problem. *European Journal of Operational Research*, 267:996 – 1002, 2018.

[24] M. Fosgerau, E. Frejinger, and A. Karlstrom. A link based network route choice model with unrestricted choice set. *Transportation Research Part B: Methodological*, 56:70–80, 2013.

[25] E. Frejinger, M. Bierlaire, and M. Ben-Akiva. Sampling of alternatives for route choice modeling. *Transportation Research Part B: Methodological*, 43(10):984–994, 2009.

[26] E. Frejinger and M. Zimmermann. Route choice and network modeling. Technical report, Encyclopedia of Transportation, 2020. Accepted for publication.

[27] M. Gendreau, G. Laporte, and I. Parent. Heuristics for the location of inspection stations on a network. *Naval Research Logistics*, 47(4):287–303, 2000.

[28] F. Gilbert, P. Marcotte, and G. Savard. A numerical study of the logit network pricing problem. *Transportation Science*, 49(3):706 – 719, 2015.

[29] S. P. Greaves and M. A. Figliozzi. Commercial vehicle tour data collection using passive gps technology: issues and potential applications. *Transportation Research Record*, 2049:158–166, 2008.

[30] F. Gzara. A cutting plane approach for bilevel hazardous material transport network design. *Operations Research Letters*, 41(1):40–46, 2013.

[31] B. Hafeez, K. Sturm, A. Kasemsarn, G. Rawling, and E. Sherman. Truck–rail intermodal connector performance evaluation: Illinois case study. *Transportation Research Record*, 2596(1):55–64, 2016.

[32] S. Hess, M. Quddus, N. Rieser-Schüssler, and A. Daly. Developing advanced route choice models for heavy goods vehicles using gps data. *Transportation Research Part E: Logistics and Transportation Review*, 77:29–44, 2015.

[33] M. J. Hodgson. A flow-capturing location-allocation model. *Geographical Analysis*, 22(3):270–279, 1990.

[34] M. W. Horner and S. Groves. Network flow-based strategies for identifying rail park-and-ride facility locations. *Socio-Economic Planning Sciences*, 41(3):255–258, 2007.

[35] J. Hunt. Calgary tour-based microsimulation of urban commercial vehicle movements case example resource paper. 2006.

[36] R. G. Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical programming*, 32(2):146–164, 1985.

[37] J. Jia, M. T. Schaub, S. Segarra, and A. R. Benson. Graph-based semi-supervised & active learning for edge flows. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 761–771, 2019.

[38] S. Jian, D. Rey, and V. Dixit. An integrated supply-demand approach to solving optimal relocations in station-based carsharing systems. *Networks and Spatial Economics*, 19(2):611–632, 2019.

[39] R. John. Maximum likelihood estimation of discrete control processes. *SIAM journal on control and optimization*, 26(5):1006–1024, 1988.

[40] B. Y. Kara and V. Verter. Designing a road network for hazardous materials transportation. *Transportation Science*, 38(2):188–196, 2004.

[41] A. Khakbaz, A. S. Nookabadi, and S. N. Shetab-bushehri. A model for locating park-and-ride facilities on urban networks based on maximizing flow capture: a case study of isfahan, iran. *Networks and Spatial Economics*, 13(1):43–66, 2012.

[42] J.-G. Kim and M. Kuby. The deviation-flow refueling location model for optimizing a network of refueling stations. *International Journal of Hydrogen Energy*, 37:5406 –

5420, 2012.

[43] M. Labbé, P. Marcotte, and G. Savard. A bilevel model of taxation and its application to optimal highway pricing problem. *Management Science*, 44:1608–1622, 1998.

[44] P. F. Laranjeiro, D. Merchán, L. A. Godoy, M. Giannotti, H. T. Yoshizaki, M. Winkenbach, and C. B. Cunha. Using gps data to explore speed patterns and temporal fluctuations in urban logistics: The case of são paulo, brazil. *Journal of Transport Geography*, 76:114–129, 2019.

[45] W. Lin and G. Hua. The flow capturing location model and algorithm of electric vehicle charging stations. *International Conference on Logistics, Informatics and Service Sciences*, 2015.

[46] T. Mai, F. Bastin, and E. Frejinger. A decomposition method for estimating recursive logit based route choice models. *EURO Journal on Transportation and Logistics*, 7(3):253–275, 2018.

[47] T. Mai, M. Fosgerau, and E. Frejinger. A nested recursive logit model for route choice analysis. *Transportation Research Part B: Methodological*, 75:100–112, 2015.

[48] P. Marcotte, A. Mercier, G. Savard, and V. Verter. Toll policies for mitigating hazardous materials transport risk. *Transportation science*, 43(2):228–243, 2009.

[49] V. Marianov, M. Rios, and M. J. Icaza. Facility location for market capture when users rank facilities by shorter travel and waiting times. *European Journal of Operational Research*, pages 30 – 42, 2008.

[50] N. Markovic, I. O. Ryzhov, and P. Schonfeld. Evasive flow capture: Optimal location of weigh-in-motion systems, tollbooths, and security checkpoints. *Networks*, 2014.

[51] N. Markovic, I. O. Ryzhov, and P. Schonfeld. Evasive flow capture: A multi-period stochastic facility location problem with independent demand. *European Journal of Operational Research*, 257(2):687–703, 2017.

[52] A. D. Marra, H. Becker, K. W. Axhausen, and F. Corman. Developing a passive gps tracking system to study long-term travel behavior. *Transportation Research Part C: Emerging Technologies*, 104:348–368, 2019.

[53] D. McDaniel and M. Devine. A modified benders' partitioning algorithm for mixed integer programming. *Management Science*, 24:241–363, 1977.

[54] D. McFadden. Modeling the choice of residential location. *Transportation Research Record*, (673), 1978.

[55] E. Murakami and D. Wagner. Can using global positioning system (gps) improve trip reporting? *Transportation Research Part C*, pages 149 – 165, 1999.

[56] M. Pacheco, S. Sharif Azadeh, M. Bierlaire, and B. Gendron. Integrating advanced discrete choice models in mixed integer linear optimization. Technical report, 2017.

[57] A. A. Prakash. Pruning algorithm for the least expected travel time path on stochastic and time-dependent networks. *Transportation Research Part B: Methodological*,

108:127–147, 2018.

[58] C. G. Prato. Route choice modeling: past, present and future research directions. *Journal of choice modelling*, 2(1):65–100, 2009.

[59] M. A. Quddus, W. Y. Ochieng, and R. B. Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation Research Part C: Emerging Technologies*, 15(5):312 – 328, 2007.

[60] R. Riemann, D. Z. Wang, and F. Busch. Optimal location of wireless charging facilities for electric vehicles: flow-capturing location model with stochastic user equilibrium. *Transportation Research Part C*, 58:1–12, 2015.

[61] N. Schuessler and K. W. Axhausen. Processing raw data from global positioning systems without additional information. *Transportation Research Record*, 2105(1):28–36, 2009.

[62] A. Sen, A. Atamturk, and P. Kaminsky. A conic integer programming approach to constrained assortment optimization under the mixed multinomial logit model. *arXiv preprint arXiv:1705.09040*, 2017.

[63] B. W. Sharman and M. J. Roorda. Multilevel modelling of commercial vehicle inter-arrival duration using gps data. *Transportation Research Part E: Logistics and Transportation Review*, 56:94–107, 2013.

[64] A. Sinha, P. Malo, and K. Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2017.

[65] P. R. Stopher, Q. Jiang, C. FitzGerald, et al. Processing gps data from travel surveys. *2nd international colloqium on the behavioural foundations of integrated land-use and transportation models: frameworks, models and applications, Toronto*, 2005.

[66] K. Sturm, Z. Pourabdollahi, A. K. Mohammadian, and K. Kawamura. Gps and driver log–based survey of grocery trucks in chicago, illinois. *Transportation Research Record*, 2410(1):31–38, 2014.

[67] Y. Sun, T. Toledo, K. Rosa, M. E. Ben-Akiva, K. Flanagan, R. Sanchez, and E. Spissu. Route choice characteristics for truckers. *Transportation Research Record*, 2354(1):115–121, 2013.

[68] A. Thakur, A. R. Pinjari, A. B. Zanjani, J. Short, V. Mysore, and S. F. Tabatabaee. Development of algorithms to convert large streams of truck gps data into truck trips. *Transportation Research Record*, 2529(1):66–73, 2015.

[69] L. Vicente, G. Savard, and J. Júdice. Descent approaches for quadratic bilevel programming. *Journal of Optimization Theory and Applications*, 81(2):379–399, 1994.

[70] D. J. White and G. Anandalingam. A penalty function approach for solving bi-level linear programs. *Journal of Global Optimization*, 3(4):397–419, 1993.

[71] F. Wu and R. Sioshansi. A stochastic flow-capturing model to optimize the location of fast-charging stations with uncertain electric vehicle flows. *Transportation Research*

Part D, 53:354–376, 2017.

[72] X. Yang, Z. Sun, X. J. Ban, and J. Holguín-Veras. Urban freight delivery stop identification with gps data. *Transportation Research Record*, 2411(1):55–61, 2014.

[73] P. P. Ypsilantis and R. R. Zuidwijk. Joint design and pricing of intermodal port-hinterland network services: Considering economies of scale and service time constraints. Technical report, 2013.

[74] W. Zeng, I. Castillo, and M. J. Hodgson. A generalized model for locating facilities on a network with flow-based demand. *Networks and Spatial Economics*, 10(4):579–611, 2008.

[75] S. Zhu, G. Amirjamshidi, and M. J. Roorda. Data fusion of commercial vehicle gps and roadside intercept survey data. *Transportation Research Record*, 2672(44):10–20, 2018.

[76] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

[77] M. Zimmermann and E. Frejinger. A tutorial on recursive models for analyzing and predicting path choice behavior. *EURO Journal on Transportation and Logistics*, 2019. Accepted for publication in November 2019. Preliminary version publicly available as ArXiv:1905.00883.

[78] M. Zimmermann, T. Mai, and E. Frejinger. Bike route choice modeling using gps data without choice sets of paths. *Transportation research part C: emerging technologies*, 75:183–196, 2017.

Appendix A

Complete Model Descriptions

A.1. Model CS

$$Z = \max \frac{1}{|S|} \sum_{s\in S} \sum_{a\in A_R} \sum_{r\in R_a} \sum_{k\in K} \sum_{l\in L} (q_{ar} d_l^k) v_{arl}^{ks}$$

$$\sum_{a\in A_R} \sum_{r\in R_a} c_{ar} y_{ar} \le b,$$

$$\sum_{r\in R_a} y_{ar} \le 1, \quad a \in A_R,$$

$$y_{ar} \in \{0,1\}, \quad a \in A, r \in R_a,$$

$$\sum_{a\in A_i^+} x_{al}^{ks} - \sum_{a\in A_i^-} x_{al}^{ks} = \begin{cases} 1, & \text{if } i = O(k), \\ -1, & \text{if } i = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad i \in N, k \in K, l \in L, s \in S,$$

$$\sum_{a\in A_n^-} x_{al}^{ks} \le 1, \quad n \in N, l \in L, k \in K, s \in S.$$

$$\pi_{jl}^{ks} - \pi_{il}^{ks} \le \mathbf{u}_{al}^{s}(y,z,\varepsilon^s), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$\pi_{O(k)l}^{ks} = 0, \quad k \in K, l \in L, s \in S,$$

$$\mathbf{u}_{al}^{s}(y,z,\varepsilon^s) - \pi_{jl}^{ks} + \pi_{il}^{ks} \le M_{als}^{k}(1 - x_{al}^{ks}), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$x_{al}^{ks} \in \{0,1\}, \quad a \in A, k \in K, l \in L, s \in S$$

$$v_{arl}^{ks} \le x_{al}^{ks}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \le y_{ar}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \ge x_{al}^{ks} - (1 - y_{ar}), \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \ge 0, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S.$$

A.2. Model CS-L

$$Z = \max \frac{1}{|S|} \sum_{s \in S} \left\{ \sum_{a \in A_R} \sum_{r \in R_a} \sum_{k \in K} \sum_{l \in L} (q_{ar} d_l^k) v_{arl}^{ks} + \sum_{k \in K} \sum_{l \in L} \lambda_l^{ks} \left(\pi_{D(k)l}^{ks} - \sum_{a \in A} \left\{ \sum_{r \in R_a} \beta_{arl} v_{arl}^{ks} + \Psi_{al}^s x_{al}^{ks} \right\} \right) \right\}$$

$$\sum_{a \in A_R} \sum_{r \in R_a} c_{ar} y_{ar} \leq b,$$

$$\sum_{r \in R_a} y_{ar} \leq 1, \quad a \in A_R,$$

$$y_{ar} \in \{0,1\}, \quad a \in A, r \in R_a,$$

$$\sum_{a \in A_i^+} x_{al}^{ks} - \sum_{a \in A_i^-} x_{al}^{ks} = \begin{cases} 1, & \text{if } i = O(k), \\ -1, & \text{if } i = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad i \in N, k \in K, l \in L, s \in S,$$

$$\sum_{a \in A_n^-} x_{al}^{ks} \leq 1, \quad n \in N, l \in L, k \in K, s \in S.$$

$$\pi_{jl}^{ks} - \pi_{il}^{ks} \leq \mathbf{u}_{al}^s(y,z,\varepsilon^s), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$\pi_{O(k)l}^{ks} = 0, \quad k \in K, l \in L, s \in S,$$

$$\mathbf{u}_{al}^s(y,z,\varepsilon^s) - \pi_{jl}^{ks} + \pi_{il}^{ks} \leq M_{als}^k (1 - x_{al}^{ks}), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$x_{al}^{ks} \in \{0,1\}, \quad a \in A, k \in K, l \in L, s \in S$$

$$v_{arl}^{ks} \leq x_{al}^{ks}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \leq y_{ar}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq x_{al}^{ks} - (1 - y_{ar}), \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq 0, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S.$$

A.3. Model SD

$$Z = \max \frac{1}{|S|} \sum_{s \in S} \sum_{a \in A_R} \sum_{r \in R_a} \sum_{k \in K} \sum_{l \in L} (q_{ar} d_l^k) v_{arl}^{ks}$$

$$\sum_{a \in A_R} \sum_{r \in R_a} c_{ar} y_{ar} \leq b,$$

$$\sum_{r \in R_a} y_{ar} \leq 1, \quad a \in A_R,$$

$$y_{ar} \in \{0,1\}, \quad a \in A, r \in R_a,$$

$$\sum_{a \in A_i^+} x_{al}^{ks} - \sum_{a \in A_i^-} x_{al}^{ks} = \begin{cases} 1, & \text{if } i = O(k), \\ -1, & \text{if } i = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad i \in N, k \in K, l \in L, s \in S,$$

$$\sum_{a \in A_n^-} x_{al}^{ks} \leq 1, \quad n \in N, l \in L, k \in K, s \in S.$$

$$x_{al}^{ks} \geq 0, \quad a \in A, k \in K, l \in L, s \in S.$$

$$\pi_{jl}^{ks} - \pi_{il}^{ks} \leq \mathbf{u}_{al}^s(y,z,\varepsilon^s), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$\pi_{O(k)l}^{ks} = 0, \quad k \in K, l \in L, s \in S,$$

$$\pi_{D(k)l}^{ks} = \sum_{a \in A} \left\{ \sum_{r \in R_a} \beta_{arl} v_{arl}^{ks} + \Psi_{al}^s x_{al}^{ks} \right\}, \quad k \in K, l \in L, s \in S.$$

$$v_{arl}^{ks} \leq x_{al}^{ks}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \leq y_{ar}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq x_{al}^{ks} - (1 - y_{ar}), \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq 0, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S.$$

A.4. Model SD-L

$$Z = \max \frac{1}{|S|} \sum_{s \in S} \left\{ \sum_{a \in A_R} \sum_{r \in R_a} \sum_{k \in K} \sum_{l \in L} (q_{ar} d_l^k) v_{arl}^{ks} + \sum_{k \in K} \sum_{l \in L} \lambda_l^{ks} \left(\pi_{D(k)l}^{ks} - \sum_{a \in A} \left\{ \sum_{r \in R_a} \beta_{arl} v_{arl}^{ks} + \Psi_{al}^s x_{al}^{ks} \right\} \right) \right\}$$

$$\sum_{a \in A_R} \sum_{r \in R_a} c_{ar} y_{ar} \leq b,$$

$$\sum_{r \in R_a} y_{ar} \leq 1, \quad a \in A_R,$$

$$y_{ar} \in \{0,1\}, \quad a \in A, r \in R_a,$$

$$\sum_{a \in A_i^+} x_{al}^{ks} - \sum_{a \in A_i^-} x_{al}^{ks} = \begin{cases} 1, & \text{if } i = O(k), \\ -1, & \text{if } i = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad i \in N, k \in K, l \in L, s \in S,$$

$$\sum_{a \in A_n^-} x_{al}^{ks} \leq 1, \quad n \in N, l \in L, k \in K, s \in S.$$

$$x_{al}^{ks} \geq 0, \quad a \in A, k \in K, l \in L, s \in S.$$

$$\pi_{jl}^{ks} - \pi_{il}^{ks} \leq \mathbf{u}_{al}^s(y,z,\varepsilon^s), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$\pi_{O(k)l}^{ks} = 0, \quad k \in K, l \in L, s \in S,$$

$$\pi_{D(k)l}^{ks} = \sum_{a \in A} \left\{ \sum_{r \in R_a} \beta_{arl} v_{arl}^{ks} + \Psi_{al}^s x_{al}^{ks} \right\}, \quad k \in K, l \in L, s \in S.$$

$$v_{arl}^{ks} \leq x_{al}^{ks}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \leq y_{ar}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq x_{al}^{ks} - (1 - y_{ar}), \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq 0, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S.$$

A.5. Model L

$$Z = \max \frac{1}{|S|} \sum_{s \in S} \left\{ \sum_{a \in A_R} \sum_{r \in R_a} \sum_{k \in K} \sum_{l \in L} (q_{ar} d_l^k) v_{arl}^{ks} + \sum_{k \in K} \sum_{l \in L} \lambda_l^{ks} \left(\pi_{D(k)l}^{ks} - \sum_{a \in A} \left\{ \sum_{r \in R_a} \beta_{arl} v_{arl}^{ks} + \Psi_{al}^s x_{al}^{ks} \right\} \right) \right\}$$

$$\sum_{a \in A_R} \sum_{r \in R_a} c_{ar} y_{ar} \leq b,$$

$$\sum_{r \in R_a} y_{ar} \leq 1, \quad a \in A_R,$$

$$y_{ar} \in \{0,1\}, \quad a \in A, r \in R_a,$$

$$\sum_{a \in A_i^+} x_{al}^{ks} - \sum_{a \in A_i^-} x_{al}^{ks} = \begin{cases} 1, & \text{if } i = O(k), \\ -1, & \text{if } i = D(k), \\ 0, & \text{otherwise,} \end{cases} \quad i \in N, k \in K, l \in L, s \in S,$$

$$\sum_{a \in A_n^-} x_{al}^{ks} \leq 1, \quad n \in N, l \in L, k \in K, s \in S.$$

$$x_{al}^{ks} \geq 0, \quad a \in A, k \in K, l \in L, s \in S.$$

$$\pi_{jl}^{ks} - \pi_{il}^{ks} \leq \mathbf{u}_{al}^s(y,z,\varepsilon^s), \quad a = (i,j) \in A, k \in K, l \in L, s \in S,$$

$$\pi_{O(k)l}^{ks} = 0, \quad k \in K, l \in L, s \in S,$$

$$v_{arl}^{ks} \leq x_{al}^{ks}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \leq y_{ar}, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq x_{al}^{ks} - (1 - y_{ar}), \quad a \in A, r \in R_a, l \in L, k \in K, s \in S,$$

$$v_{arl}^{ks} \geq 0, \quad a \in A, r \in R_a, l \in L, k \in K, s \in S.$$

Appendix B

Joint Network Design and Pricing

Joint network design and pricing (JNDP), unlike network pricing, does not enjoy a vast literature. The combination of network design and pricing aspects within the same model is challenging which makes it an interesting subject for our method. We provide both a literature review and detailed mathematical developments leading to our Benders decomposition for the problem.

B.1. Literature Review

Joint network design and pricing can be seen from a similar perspective as the HTNDP: the leader's problem consists of maximizing profits by deciding which links can be used by the users and at what cost, while the follower's problem remains to find the shortest path, from origin to destination, in regards to several factors including the tariffs. The same sequential and uncooperative interaction between leader and follower exists in this problem as it does in the hazmat problem.

The literature surrounding the joint network design and pricing problem remains fairly scarce despite the relatively large body of work on network design problems and pricing problems respectively. To the best of our knowledge, the first solution method for the JNDP is introduced in [13] where a comparison between a heuristic and the exact bilevel model reformulated to a MILP solved by CPLEX is provided. We note that the idea of moving the strong duality constraint to the objective function along with a weight is used in the heuristic. It also decomposes the subproblem in shortest path problems by commodity. Their work is extended in [14] where different levels of capacity are considered. A similar comparison is provided which shows the heuristic performing well whereas the performance of the MIP is relatively variable.

[73] propose a model which takes into account service time constraints and economies of scale. Similarly to [13] and [14], a heuristic is proposed and is shown to provide near optimal solutions in relatively short times.

B.2. Applying the Decomposition

The bilevel model we use as a starting point is a straightforward adaptation of the one found in [13] where the continuous tariff variables are changed to integer variables each multiplying a different possible tariff value. We argue that this is a reasonable change which does not entail any loss of generality as monetary quantities fundamentally take on integer values.

We denote the graph representing the full network as $G = (N,A)$ and A_1 as the set of arcs controlled by the leader and A_2 as $A \backslash A_1$. Binary variables y_{ij} indicate whether an arc $a = (i,j) \in A_1$ is open or not. A fixed cost f_{ij} is associated with the opening of an arc $a \in A_1$ as well as an operating cost c_{ij} which is multiplied by the quantity of flow d^k using the link across all OD pairs $k \in K$. Binary variables t_{ij}^{τ} indicate which tariff T^{τ} is selected from the set of possible tariffs $\mathscr{T}$ for a given arc $a \in A_1$. For arcs $a \in A_2$, competitors are assumed to impose a certain tariff u_{ij}. Flow variables x_{ij}^k indicate which arcs $(i,j) \in A_1$ are used by users of OD pair k and flow variables s_{ij}^k are homologous with A_2.

$$\max \sum_{k \in K} \sum_{(i,j) \in A_1} \sum_{\tau \in \mathscr{T}} T^{\tau} t_{ij}^{\tau} d^k x_{ij}^k - \sum_{(i,j) \in A_1} f_{ij} y_{ij} - \sum_{k \in K} \sum_{(i,j) \in A_1} c_{ij} d^k x_{ij}^k, \tag{B.2.1}$$

$$\sum_{\tau \in \mathscr{T}} t_{ij}^{\tau} \leq 1 \qquad \forall (i,j) \in A_1, \tag{B.2.2}$$

$$t_{ij}^{\tau} \in \{0,1\} \qquad \forall (i,j) \in A_1, \forall \tau \in \mathscr{T}, \tag{B.2.3}$$

$$y_{ij} \in \{0,1\} \qquad \forall (i,j) \in A_1, \tag{B.2.4}$$

$$\min \sum_{k \in K} d^k \Big(\sum_{(i,j) \in A_1} \sum_{\tau \in \mathscr{T}} T^{\tau} t_{ij}^{\tau} x_{ij}^k + \sum_{(i,j) \in A_2} u_{ij} s_{ij}^k \Big), \tag{B.2.5}$$

$$x_{ij}^k \leq y_{ij} \qquad \forall (i,j) \in A_1, \forall k \in K, \tag{B.2.6}$$

$$Ex^k + Fs^k = \begin{cases} -1, & \text{if } j = O(k), \\ 1, & \text{if } j = D(k), \\ 0, & \text{otherwise,} \end{cases} \qquad \forall j \in N, k \in K, \tag{B.2.7}$$

$$x_{ij}^k \geq 0 \qquad \forall (i,j) \in A_1, \tag{B.2.8}$$

$$s_{ij}^k \geq 0 \qquad \forall (i,j) \in A_2, \tag{B.2.9}$$

In constraint (B.2.7), E and F represent coefficient matrices for flow conservation constraints. The leader's objective (B.2.1) consists of maximizing profits from the imposed tariffs while minimizing the fixed costs and operational costs of opening arcs. Constraints (B.2.2) implies that only one tariff can be selected at most for each arc $\in A_1$. The follower's objective (B.2.5) consists of minimizing the costs incurred by using the network (the tariffs, in this case). Constraints (B.2.6) are the classical flow conservation constraints adapted to the two subsets of arcs A_1 and A_2.

As was the case in the original formulation of the HTNDP, (B.2.6) must be reformulated as a penalty in the objective function of the follower and the problem as a whole can be separated by commodity k. The resulting formulation corresponds to the form (1.1)-(1.3)[1]. Therefore, we can apply our Benders decomposition method as long as (5)-(6) are verified. Assumption (5) is verified since our tariffs represented by discrete variables, therefore an upper bound can be calculated by considering only the highest tariffs possible. In the formulation of [13], tariffs are continuous and ≥ 0 and so it is assumed that there exists at least one tariff-free path for each OD pair. We believe our formulation does not require this assumption. Assumption (6) holds because arc travel costs are assumed to be ≥ 0.

Applying our methodology leads us to solve two shortest path problems, for each $k \in K$, with the following arc costs in order to generate cuts on w and Π respectively with an integer $\overline{y}$ current solution:

$$\lambda^k\Big(d^k(\sum_{\tau\in\mathscr{T}} T^\tau t^\tau_{ij} + M(1-y_{ij}) + u_{ij})\Big) + d^k c_{ij} - d^k \sum_{\tau\in\mathscr{T}} T^\tau t^\tau_{ij} \tag{B.2.10}$$

and

$$d^k(\sum_{\tau\in\mathscr{T}} T^\tau t^\tau_{ij} + M(1-y_{ij}) + u_{ij}). \tag{B.2.11}$$

We have the following $|K|$ problems to solve for cuts on w with a fractional $\overline{y}$:

$$\begin{aligned}\max \pi_{D(k)} + \sum_{(i,j)\in A_1} \gamma^{k(y)}_{ij} y_{ij} + \sum_{(i,j)\in A} \gamma^{k(1-y)}_{ij}(1-y_{ij}) \\ + \sum_{(i,j)\in A_1} \theta^{k(t)}_{ij} t^\tau_{ij} + \sum_{(i,j)\in A} \theta^{k(1-t)}_{ij}(1-t^\tau_{ij}),\end{aligned} \tag{B.2.12}$$

$$E^T\pi^k - F^T\pi^k - \gamma^k_{ij} + \gamma^{k(1-y)}_{ij} - \theta^k_{ij} + \theta^{k(1-t)}_{ij} \leq \lambda^k\Big(d^k(M+u_{ij})\Big) - d^k c_{ij} \tag{B.2.13}$$

$$\forall (i,j) \in A,$$

$$\gamma^k_{ij} + \gamma^{k(y)}_{ij} - \gamma^{k(1-y)}_{ij} \geq -\lambda M \qquad \forall (i,j) \in A, \tag{B.2.14}$$

$$\gamma^k_{ij}, \gamma^{k(y)}_{ij}, \gamma^{k(1-y)}_{ij} \geq 0 \qquad \forall (i,j) \in A, \tag{B.2.15}$$

$$\theta^k_{ij} + \theta^{k(t)}_{ij} - \theta^{k(1-t)}_{ij} \geq (\lambda - 1) d^k \sum_{\tau\in\mathscr{T}} T^\tau \qquad \forall (i,j) \in A, \tag{B.2.16}$$

$$\theta^k_{ij}, \theta^{k(t)}_{ij}, \theta^{k(1-t)}_{ij} \geq 0 \qquad \forall (i,j) \in A. \tag{B.2.17}$$

Finally, we can write our Benders reformulation for the hazmat transportation network design problem as:

$$\max -\sum_{ij\in A_1} f_{ij} y_{ij} + \sum_{k\in K} w^k - \lambda^k \Pi^k \tag{B.2.18}$$

[1]The references in this Appendix are found in the third article.

subject to (B.2.2)-(B.2.4) and

$$\begin{aligned} w^k \leq \pi_{D(k)} + \sum_{(i,j)\in A_1} \gamma_{ij}^{k(y)} y_{ij} + \sum_{(i,j)\in A} \gamma_{ij}^{k(1-y)}(1-y_{ij}) \\ + \sum_{(i,j)\in A_1} \theta_{ij}^{k(t)} t_{ij}^{\tau} + \sum_{(i,j)\in A} \theta_{ij}^{k(1-t)}(1-t_{ij}^{\tau}) \end{aligned} \tag{B.2.19}$$

$$\forall k \in K, \pi_i^k, \gamma_{ij}^k, \gamma_{ij}^{k(y)}, \gamma_{ij}^{k(1-y)}, \theta_{ij}^k + \theta_{ij}^{k(t)} - \theta_{ij}^{k(1-t)} \in \mathscr{D}_1^k,$$

$$\Pi^k \geq d^k \Big(\sum_{(i,j)\in A_1} \sum_{\tau\in\mathscr{T}} T^{\tau} t_{ij}^{\tau} x_{ij}^k + \sum_{(i,j)\in A_2} u_{ij} s_{ij}^k + \sum_{(i,j)\in A_1} M(1-y_{ij}) x_{ij}^k \Big) \tag{B.2.20}$$

$$\forall k \in K, x, s \in \mathscr{D}_2^k,$$

where $\mathscr{D}_1^k$ is the solution space of (B.2.12)-(B.2.17) and $\mathscr{D}_2^k$ is the solution space of (B.2.5)-(B.2.9) when (B.2.6) is rewritten as a penalty in the objective function.